

This Page Is Inserted by IFW Operations
and is not a part of the Official Record

BEST AVAILABLE IMAGES

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images may include (but are not limited to):

- BLACK BORDERS
- TEXT CUT OFF AT TOP, BOTTOM OR SIDES
- FADED TEXT
- BLURRY OR ILLEGIBLE TEXT
- SKEWED/SLATED IMAGES
- COLORED PHOTOS
- BLACK OR VERY DARK BLACK AND WHITE PHOTOS
- UNDECIPHERABLE GRAY SCALE DOCUMENTS

IMAGES ARE BEST AVAILABLE COPY.

**As rescanning documents *will not* correct images,
please do not report the images to the
Image Problem Mailbox.**

This Page Blank (uspto)

日本国特許庁

PATENT OFFICE
JAPANESE GOVERNMENT

PCT/JP00/C4637

JP00/4637

12.07.00

別紙添付の書類に記載されている事項は下記の出願書類に記載されている事項と同一であることを証明する。

This is to certify that the annexed is a true copy of the following application as filed with this Office.

出願年月日

Date of Application:

1999年 7月15日

出願番号

Application Number:

平成11年特許願第201248号

出願人

Applicant(s):

セイコーエプソン株式会社

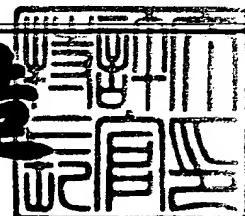
PRIORITY
DOCUMENT

SUBMITTED OR TRANSMITTED IN
COMPLIANCE WITH RULE 17.1(a) OR (b)

2000年 8月18日

特許庁長官
Commissioner,
Patent Office

及川耕造



出証番号 出証特2000-3064448

【書類名】 特許願
 【整理番号】 EP192001
 【提出日】 平成11年 7月15日
 【あて先】 特許庁長官殿
 【国際特許分類】 G06F 13/12
 【発明者】

【住所又は居所】 長野県諏訪市大和3丁目3番5号 セイコーエプソン株式会社内

【氏名】 石田 卓也

【発明者】

【住所又は居所】 長野県諏訪市大和3丁目3番5号 セイコーエプソン株式会社内

【氏名】 神原 義幸

【特許出願人】

【識別番号】 000002369

【氏名又は名称】 セイコーエプソン株式会社

【代理人】

【識別番号】 100090479

【弁理士】

【氏名又は名称】 井上 一

【電話番号】 03-5397-0891

【選任した代理人】

【識別番号】 100090387

【弁理士】

【氏名又は名称】 布施 行夫

【電話番号】 03-5397-0891

【選任した代理人】

【識別番号】 100090398

【弁理士】

【氏名又は名称】 大 瀧 美千栄

【電話番号】 03-5397-0891

【手数料の表示】

【予納台帳番号】 039491

【納付金額】 21,000円

【提出物件の目録】

【物件名】 明細書 1

【物件名】 図面 1

【物件名】 要約書 1

【包括委任状番号】 9402500

【プルーフの要否】 要

【書類名】 明細書

【発明の名称】 データ転送制御装置及び電子機器

【特許請求の範囲】

【請求項 1】 バスに接続される複数のノード間でのデータ転送のためのデータ転送制御装置であって、

ノードのトポロジ情報をクリアするリセットから次のリセットまでの間をリセットインターバルと定義した場合に、受信したパケットと次に受信したパケットとが異なるリセットインターバルに受信したパケットか否かを区別するための区別情報を生成する手段と、

受信した各パケットと生成された各区別情報とを、各パケットに各区別情報を関連づけて、パケット記憶手段に書き込む書き込み手段と、

を含むことを特徴とするデータ転送制御装置。

【請求項 2】 請求項 1 において、
前記区別情報が、

受信したパケットと次に受信したパケットとが異なるリセットインターバルに受信したパケットである場合に、0 から 1 に或いは 1 から 0 にトグルするトグルビットであることを特徴とするデータ転送制御装置。

【請求項 3】 請求項 1 又は 2 において、
前記パケット記憶手段がランダムアクセス可能な記憶手段であり、前記パケット記憶手段が、パケットの制御情報が格納される制御情報領域とパケットのデータが格納されるデータ領域とに分離されている場合において、

前記区別情報を、前記制御情報領域に書き込まれる前記制御情報の中に含ませることを特徴とするデータ転送制御装置。

【請求項 4】 バスに接続される複数のノード間でのデータ転送のためのデータ転送制御装置であって、

各ノードから受信したパケットをパケット記憶手段に書き込む書き込み手段と

ノードのトポロジ情報をクリアするリセットの発生前の受信パケットの領域と

リセットの発生後の受信パケットの領域との、前記パケット記憶手段における境界を特定する第1のポインタ情報を記憶する第1のポインタ記憶手段と、
を含むことを特徴とするデータ転送制御装置。

【請求項5】 請求項4において、

リセットの発生の直前に受信したパケットの次のパケットの先頭アドレスが、前記第1のポインタ情報として前記第1のポインタ記憶手段に記憶されることを特徴とするデータ転送制御装置。

【請求項6】 請求項4又は5において、

処理済みパケットの領域と未処理のパケットの領域との、前記パケット記憶手段における境界を特定する第2のポインタ情報を記憶する第2のポインタ記憶手段と、

受信済みパケットの領域とパケットを未受信の領域との、前記パケット記憶手段における境界を特定する第3のポインタ情報を記憶する第3のポインタ記憶手段とを含むことを特徴とするデータ転送制御装置。

【請求項7】 請求項4乃至6のいずれかにおいて、

前記第1のポインタ記憶手段に記憶される前記第1のポインタ情報に基づいてリセット発生後の受信パケットを特定し、該パケットを優先的に処理する処理手段を含むことを特徴とするデータ転送制御装置。

【請求項8】 請求項4乃至7のいずれかにおいて、

前記パケット記憶手段がランダムアクセス可能な記憶手段であり、前記パケット記憶手段が、パケットの制御情報が格納される制御情報領域とパケットのデータが格納されるデータ領域とに分離されている場合において、

前記第1のポインタ記憶手段が、

ノードのトポロジ情報をクリアするリセットの発生前の受信パケットの制御情報とリセット発生後の受信パケットの制御情報との、前記制御情報領域における境界を特定する第4のポインタ情報を記憶する第4のポインタ記憶手段と、

ノードのトポロジ情報をクリアするリセットの発生前の受信パケットのデータとリセット発生後の受信パケットのデータとの、前記データ領域における境界を特定する第5のポインタ情報を記憶する第5のポインタ記憶手段とを含むことを

特徴とするデータ転送制御装置。

【請求項 9】 請求項 8 において、

前記データ領域が、第 1 の層用の第 1 のデータが格納される第 1 のデータ領域と、第 2 の層用の第 2 のデータが格納される第 2 のデータ領域とに分離されている場合において、

前記第 5 のポインタ情報が、

ノードのトポロジ情報をクリアするリセットの発生前の受信パケットの第 1 のデータとリセットの発生後の受信パケットの第 1 のデータとの、前記第 1 のデータ領域における境界を特定するポインタ情報であることを特徴とするデータ転送制御装置。

【請求項 10】 バスに接続される複数のノード間でのデータ転送のためのデータ転送制御装置であって、

送信開始コマンドが発行された場合に、パケット記憶手段からパケットを読み出す読み出し手段と、

読み出されたパケットを各ノードに送信するためのサービスを提供するリンク手段と、

ノードのトポロジ情報をクリアするリセットの発生によりパケットの送信が中止された場合に、リセットの発生によりパケットの送信が中止されたことを知らせるステータス情報を記憶するステータス記憶手段と、

を含むことを特徴とするデータ転送制御装置。

【請求項 11】 請求項 10 において、

前記送信開始コマンドを発行する処理手段を含み、

前記処理手段が、

リセットの発生によりパケットの送信が中止されたと前記ステータス情報に基づいて判断した場合には、送信が完了したか否かを判断することなく、開始した送信処理をキャンセルすることを特徴とするデータ転送制御装置。

【請求項 12】 請求項 1 乃至 11 のいずれかにおいて、

前記リセットが、IEEE 1394 の規格において定義されるバスリセットであることを特徴とするデータ転送制御装置。

【請求項 1 3】 請求項 1 乃至 1 2 のいずれかにおいて、
I E E E 1 3 9 4 の規格に準拠したデータ転送を行うことを特徴とするデータ
転送制御装置。

【請求項 1 4】 請求項 1 乃至 1 3 のいずれかのデータ転送制御装置と、
前記データ転送制御装置及びバスを介して他のノードから受信したデータに所
与の処理を施す装置と、

処理が施されたデータを出力又は記憶するための装置とを含むことを特徴とす
る電子機器。

【請求項 1 5】 請求項 1 乃至 1 3 のいずれかのデータ転送制御装置と、
前記データ転送制御装置及びバスを介して他のノードに送信するデータに所与
の処理を施す装置と、

処理が施されるデータを取り込むための装置とを含むことを特徴とする電子機
器。

【発明の詳細な説明】

【0 0 0 1】

【発明の属する技術分野】

本発明は、データ転送制御装置及びこれを含む電子機器に関し、特に、バスに
接続される複数のノード間で I E E E 1 3 9 4 などの規格に準じたデータ転送を
行うデータ転送制御装置及びこれを含む電子機器に関する。

【0 0 0 2】

【背景技術及び発明が解決しようとする課題】

近年、I E E E 1 3 9 4 と呼ばれるインターフェース規格が脚光を浴びている
。この I E E E 1 3 9 4 は、次世代のマルチメディアにも対応可能な高速シリアル
バスインターフェースを規格化したものである。この I E E E 1 3 9 4 によれ
ば、動画像などのリアルタイム性が要求されるデータも扱うことができる。また

、I E E E 1 3 9 4 のバスには、プリンタ、スキャナ、C D - R W ドライブ、ハ
ードディスクドライブなどのコンピュータの周辺機器のみならず、ビデオカメラ
、V T R、T V などの家庭用電化製品も接続できる。このため、電子機器のデジ
タル化を飛躍的に促進できるものとして期待されている。

【0003】

さて、このIEEE 1394においては、バスに電子機器が新たに接続されたり、バスから電子機器が取り外されたりして、バスに接続されるノードが増減すると、いわゆるバスリセットが発生する。そしてバスリセットが発生するとノードのトポロジ情報がクリアされ、その後、トポロジ情報が自動的に再設定される。即ち、バスリセットの発生後、ツリー識別（ルートノードの決定）、自己識別が行われ、その後、アイソクロナスリソースマネージャ等の管理ノードが決定される。そして通常の packets 転送が開始される。

【0004】

このようにIEEE 1394では、バスリセット後にトポロジ情報が自動的に再設定されるため、いわゆるホット状態でのケーブルの抜き差し（ホットプラグ）が可能となる。このため、一般ユーザは、VTRなどの通常の家庭用電化製品と同じように、電子機器へのケーブルの抜き差しを自由にできるようになり、いわゆるホームネットワークシステムの普及に役立つことができる。

【0005】

しかしながら、このようなバスリセットには、以下のような問題があることが判明した。

(1) バスリセットの発生によりノードID等のトポロジ情報がクリアされるため、トランザクションの途中でバスリセットが発生した場合には、トランザクションを再度やり直す必要がある。従って、各ノードは、バスリセット発生時に、どのトランザクションが未完了であったかを判断する必要がある。

【0006】

しかしながら、パケットを処理するファームウェア（CPU）の処理能力は一般的に低く、受信パケットに対する処理は、そのパケットの受信後、所与の時間が経過してから行われる。このため、多くの未処理のパケットが常に存在しており、これらの多くの未処理のパケットが、バスリセットの前後のいずれに受信したパケットなのかを区別する処理が必要になる。ところが、このような処理はファームウェアにとって負担が非常に重い。特に、バスリセット後にはノード間で非常に多くのパケットが送受信されるため、バスリセット後に受信するパケット

の数は非常に多く、ファームウェアの処理負担の増加の問題は更に深刻なものとなる。

(2) ファームウェアがパケットの送信開始コマンドを発行した後にバスリセットが発生した場合、その送信は中止される。ところが、例えば送信開始コマンドの発行とバスリセットとがほぼ同時に起きた場合には、バスリセットの発生により送信が中止したか否かをファームウェアは知ることができない。このため、ファームウェアは、送信完了のステータスが送られてくるまで待機することになってしまい、ファームウェアの処理がストールするおそれがある。

【0007】

本発明は、以上のような技術的課題に鑑みてなされたものであり、その目的とするところは、ノードのトポロジ情報をクリアするリセットの発生時のファームウェアの処理負担を軽減でき、リセット発生によるファームウェアの処理のストール等を防止できるデータ転送制御装置及びこれが用いられる電子機器を提供することにある。

【0008】

【課題を解決するための手段】

上記課題を解決するために本発明は、バスに接続される複数のノード間でのデータ転送のためのデータ転送制御装置であって、ノードのトポロジ情報をクリアするリセットから次のリセットまでの間をリセットインターバルと定義した場合に、受信したパケットと次に受信したパケットとが異なるリセットインターバルに受信したパケットか否かを区別するための区別情報を生成する手段と、受信した各パケットと生成された各区別情報とを、各パケットに各区別情報を関連づけて、パケット記憶手段に書き込む書き込み手段とを含むことを特徴とする。

【0009】

本発明によれば、例えば、受信した第Nのパケットに関連づけられた第Nの区別情報と、次に受信した第N+1のパケットに関連づけられた第N+1の区別情報とにより、第N、N+1のパケットが異なるリセットインターバルに受信されたか否かを区別できるようになる。即ち、第N、N+1のパケットが、同一の第Mのリセットインターバルに受信された場合には、第N、第N+1の区別情報は

例えば同じ値になり、第N、N+1のパケットが、異なる第M、第Lのリセットインターバルに受信された場合には、第N、第N+1の区別情報は例えば異なる値になる。従って、第N、第N+1の区別情報の変化を調べることで、第N、第N+1のパケットが異なるリセットインターバルに受信されたか否か、即ち第N、第N+1のパケットの受信の間にリセットが発生したか否を判別できるようになる。従って、ファームウェア等は、リセットの発生場所を容易に知ることができるようになり、リセット発生時の処理負担を軽減できるようになる。

【0010】

また本発明は、前記区別情報が、受信したパケットと次に受信したパケットとが異なるリセットインターバルに受信したパケットである場合に、0から1に或いは1から0にトグルするトグルビットであることを特徴とする。このようにすれば、トグルビットの変化を調べるだけで、リセットの発生場所を容易に知ることができるようになり、リセット発生時のファームウェア等の処理負担を更に軽減できるようになる。

【0011】

また本発明は、前記パケット記憶手段がランダムアクセス可能な記憶手段であり、前記パケット記憶手段が、パケットの制御情報が格納される制御情報領域とパケットのデータが格納されるデータ領域とに分離されている場合において、前記区別情報を、前記制御情報領域に書き込まれる前記制御情報の中に含ませることを特徴とする。このように、パケット記憶手段を制御情報領域とデータ領域に分離することで、ファームウェア等の処理負担を軽減でき、システム全体の実転送速度を向上できる。そして、区別情報を、制御情報領域に書き込まれる制御情報の中に含ませることで、ファームウェア等は、少ない処理負担で区別情報を容易に読み出すことができるようになる。

【0012】

また本発明は、バスに接続される複数のノード間でのデータ転送のためのデータ転送制御装置であって、各ノードから受信したパケットをパケット記憶手段に書き込む書き込み手段と、ノードのトポロジ情報をクリアするリセットの発生前の受信パケットの領域とリセットの発生後の受信パケットの領域との、前記パケ

ット記憶手段における境界を特定する第1のポインタ情報を記憶する第1のポインタ記憶手段とを含むことを特徴とする。

【0013】

本発明によれば、ファームウェア等は、第1のポインタ記憶手段から第1のポインタ情報を読み出すだけで、リセット発生前に受信したパケットとリセット発生後に受信したパケットとを容易に区別できるようになる。そして、リセット発生前に受信したパケットに対しては破棄する等の処理を行い、リセット発生後に受信したパケットに対しては通常の処理を行うこと等が可能になる。

【0014】

また本発明は、リセットの発生の直前に受信したパケットの次のパケットの先頭アドレスが、前記第1のポインタ情報として前記第1のポインタ記憶手段に記憶されることを特徴とする。このようにすれば、第1のポインタ記憶手段から第1のポインタ情報を読み出すだけで、リセットの発生の直前に受信したパケットの次のパケットに対する処理を開始できるようになり、ファームウェア等の処理負担を軽減できる。

【0015】

また本発明は、処理済みパケットの領域と未処理のパケットの領域との、前記パケット記憶手段における境界を特定する第2のポインタ情報を記憶する第2のポインタ記憶手段と、受信済みパケットの領域とパケットを未受信の領域との、前記パケット記憶手段における境界を特定する第3のポインタ情報を記憶する第3のポインタ記憶手段とを含むことを特徴とする。このようにすれば、第2、第3のポインタ記憶手段から第2、第3のポインタ情報を読み出すだけで、未処理のパケットや、最新に受信したパケットなどを容易に判別できるようになる。

【0016】

また本発明は、前記第1のポインタ記憶手段に記憶される前記第1のポインタ情報に基づいてリセット発生後の受信パケットを特定し、該パケットを優先的に処理する処理手段を含むことを特徴とする。このようにすれば、リセット発生後の受信パケットが優先的に処理されるようになり、ファームウェアの処理がストールするなどの事態が生じるのを防止できるようになる。

【0017】

また本発明は、前記パケット記憶手段がランダムアクセス可能な記憶手段であり、前記パケット記憶手段が、パケットの制御情報が格納される制御情報領域とパケットのデータが格納されるデータ領域とに分離されている場合において、前記第1のポインタ記憶手段が、ノードのトポロジ情報をクリアするリセットの発生前の受信パケットの制御情報とリセット発生後の受信パケットの制御情報との、前記制御情報領域における境界を特定する第4のポインタ情報を記憶する第4のポインタ記憶手段と、ノードのトポロジ情報をクリアするリセットの発生前の受信パケットのデータとリセット発生後の受信パケットのデータとの、前記データ領域における境界を特定する第5のポインタ情報を記憶する第5のポインタ記憶手段とを含むことを特徴とする。このようにすれば、第5のポインタ記憶手段から第5のポインタ情報を読み出すだけで、リセット発生前後の受信パケットの、データ領域での境界を知ることができるようになり、ファームウェア等の処理負担を更に軽減できる。

【0018】

また本発明は、前記データ領域が、第1の層用の第1のデータが格納される第1のデータ領域と、第2の層用の第2のデータが格納される第2のデータ領域とに分離されている場合において、前記第5のポインタ情報が、ノードのトポロジ情報をクリアするリセットの発生前の受信パケットの第1のデータとリセットの発生後の受信パケットの第1のデータとの、前記第1のデータ領域における境界を特定するポインタ情報であることを特徴とする。このようにすれば、第1の層（例えばファームウェア）は、第1のデータ領域に格納される第1のデータ（例えばORB）を、リセットの発生の前後を区別して読み出すことができるようになり、第1の層の処理負担を軽減できる。また、第2の層（例えばアプリケーション）は、第2のデータ領域に格納される第2のデータ（例えばストリーム）を連続して読み出すことができるようになり、データ転送制御装置の実転送速度を向上できるようになる。

【0019】

また本発明は、バスに接続される複数のノード間でのデータ転送のためのデー

タ転送制御装置であって、送信開始コマンドが発行された場合に、パケット記憶手段からパケットを読み出す読み出し手段と、読み出されたパケットを各ノードに送信するためのサービスを提供するリンク手段と、ノードのトポロジ情報をクリアするリセットの発生によりパケットの送信が中止された場合に、リセットの発生によりパケットの送信が中止されたことを知らせるステータス情報を記憶するステータス記憶手段とを含むことを特徴とする。

【0020】

本発明によれば、ファームウェア等は、ステータス記憶手段からステータス情報を読み出すことで、リセットの発生によりパケットの送信が中止されたことを知ることができるようになる。この結果、例えば、送信開始コマンドの発行の少し前にリセットが発生した場合等においても、ファームウェア等の処理がストールしてしまう事態を防止できるようになる。

【0021】

また本発明は、前記送信開始コマンドを発行する処理手段を含み、前記処理手段が、リセットの発生によりパケットの送信が中止されたと前記ステータス情報に基づいて判断した場合には、送信が完了したか否かを判断することなく、開始した送信処理をキャンセルすることを特徴とする。このようにすれば、ファームウェア等は、送信完了のステータスが送られてくるのを待つ必要がなくなり、ファームウェア等の処理がストールする事態を防止できる。

【0022】

また本発明では、前記リセットが、IEEE 1394の規格において定義されるバスリセットであることが望ましい。

【0023】

また、本発明では、IEEE 1394の規格に準拠したデータ転送を行うことが望ましい。

【0024】

また本発明に係る電子機器は、上記のいずれかのデータ転送制御装置と、前記データ転送制御装置及びバスを介して他のノードから受信したデータに所与の処理を施す装置と、処理が施されたデータを出力又は記憶するための装置とを含む

ことを特徴とする。また本発明に係る電子機器は、上記のいずれかのデータ転送制御装置と、前記データ転送制御装置及びバスを介して他のノードに送信するデータに所与の処理を施す装置と、処理が施されるデータを取り込むための装置とを含むことを特徴とする。

【0025】

本発明によれば、データ転送を制御するファームウェア等の処理負担を軽減できるため、電子機器の低コスト化、処理の高速化などを図ることができる。また、トポロジー情報をクリアするリセットの発生によりシステムがストールしてしまうなどの事態を防止できるため、電子機器の信頼性を高めることができる。

【0026】

【発明の実施の形態】

以下、本発明の好適な実施形態について図面を用いて詳細に説明する。

【0027】

1. IEEE1394

まず、IEEE1394について簡単に説明する。

【0028】

1. 1 概要

IEEE1394 (IEEE1394-1995、P1394. a) では100～400Mbpsの高速なデータ転送が可能となっている (P1394. bでは800～3200Mbps)。また、転送速度が異なるノードをバスに接続することも許される。

【0029】

各ノードはツリー状に接続されており、1つのバスに最大で63個のノードが接続可能になっている。なお、バスブリッジを利用すれば約64000個のノードを接続することも可能である。

【0030】

IEEE1394では、パケットの転送方式として非同期転送とアイソクロナス転送が用意されている。ここで非同期転送は、信頼性が要求されるデータの転送に好適な転送方式であり、アイソクロナス転送は、リアルタイム性が要求され

る動画像や音声などのデータの転送に好適な転送方式である。

【0031】

1. 2 層構造

IEEE 1394 の層構造（プロトコル構成）を図 1 に示す。

【0032】

IEEE 1394 のプロトコルは、トランザクション層、リンク層、物理層により構成される。また、シリアルバスマネージメントは、トランザクション層、リンク層、物理層をモニターしたり制御したりするものであり、ノードの制御やバスのリソース管理のための種々の機能を提供する。

【0033】

トランザクション層は、上位層にトランザクション単位のインターフェース（サービス）を提供し、下層のリンク層が提供するインターフェースを通して、リードトランザクション、ライトトランザクション、ロックトランザクション等のトランザクションを実施する。

【0034】

ここで、リードトランザクションでは、応答ノードから要求ノードにデータが転送される。一方、ライトトランザクションでは、要求ノードから応答ノードにデータが転送される。またロックトランザクションでは、要求ノードから応答ノードにデータが転送され、応答ノードがそのデータに処理を施して要求ノードに返信する。

【0035】

トランザクション層のサービスは、図 2（A）に示すように要求、表示、応答、確認という 4 つのサービスにより構成される。

【0036】

ここで、トランザクション要求は、要求側がトランザクションを開始させるサービスであり、トランザクション表示は、要求が届いたことを応答側に通知するサービスである。また、トランザクション応答は、応答側の状態やデータを要求側に返すサービスであり、トランザクション確認は、応答側からの応答がきたことを要求側に通知するサービスである。

【0037】

リンク層は、アドレッシング、データチェック、パケット送受信のためのデータフレーミング、アイソクロナス転送のためのサイクル制御などを提供する。

【0038】

リンク層のサービスは、トランザクション層と同様に、図2(B)に示すように要求、表示、応答、確認という4つのサービスにより構成される。

【0039】

ここで、リンク要求は、パケットを応答側に転送するサービスであり、リンク表示は、応答側によるパケットの受信サービスである。また、リンク応答は、応答側によるアクノリッジメントの転送サービスであり、リンク確認は、要求側によるアクノリッジメントの受信サービスである。

【0040】

物理層は、リンク層により使用されるロジカルシンボルの電気信号への変換や、バスの調停や、バスの物理的インターフェースを提供する。

【0041】

物理層及びリンク層は、通常、データ転送制御装置（インターフェースチップ）などのハードウェアにより実現される。また、トランザクション層は、CPU上で動作するファームウェア（処理手段）や、ハードウェアにより実現される。

【0042】

なお、図3に示すように、IEEE1394のトランザクション層の一部の機能を含む上位のプロトコルとして、SBP-2 (Serial Bus Protocol-2) と呼ばれるプロトコルが提案されている。

【0043】

ここでSBP-2は、SCSIのコマンドセットをIEEE1394のプロトコル上で利用可能にするために提案されたものである。このSBP-2を用いれば、既存のSCSI規格の電子機器で使用されていたSCSIのコマンドセットに最小限の変更を加えて、IEEE1394規格の電子機器に使用できるようになる。従って、電子機器の設計や開発を容易化できる。また、SCSIのコマンドだけではなく、デバイス固有のコマンドもカプセル化して利用できるため、非常に

汎用性が高い。

【0044】

このSBP-2では、まず、イニシエータ（パーソナルコンピュータ等）が、ロ
グインやフェッチ・エージェントの初期化のためのORB（Operation Request
Block）を作成して、ターゲット（プリンタ、CD-RWドライブ等）に送る。
次に、イニシエータは、コマンド（リードコマンド、ライトコマンド）を含むO
RB（コマンドブロックORB）を作成して、その作成したORBのアドレスを
、ターゲットに知らせる。そして、ターゲットは、そのアドレスをフェッチする
ことにより、イニシエータが作成したORBを取得する。ORBに含まれるコマ
ンドがリードコマンドであった場合には、ターゲットは、ブロックライトラン
ザクションを発行して、イニシエータのデータバッファ（メモリ）にターゲット
のデータを送信する。一方、ORBに含まれるコマンドがライトコマンドであっ
た場合には、ターゲットは、ブロックリードランザクションを発行して、イニ
シエータのデータバッファからデータを受信する。

【0045】

このSBP-2によれば、ターゲットは、自身が都合の良いときにランザクシ
ョンを発行して、データを送受信できる。従って、イニシエータとターゲットが
同期して動く必要がなくなるため、データ転送効率を高めることができる。

【0046】

なお、IEEE1394の上位プロトコルとしては、SBP-2以外にも、FC
P（Function Control Protocol）と呼ばれるプロトコルなども提案されている

【0047】

1. 3 バスリセット

IEEE1394では、電源が投入されたり、途中でデバイスの抜き差しが発
生すると、バスリセットが発生する。即ち、各ノードは、ポートの電圧変化を監
視している。そして、バスに新たなノードが接続されるなどしてポートの電圧に
変化が生じると、この変化を検知したノードは、バス上の他のノードに対して、
バスリセットが発生したことを知らせる。また、各ノードの物理層は、バスリセ

ットが発生したことをリンク層に伝える。

【0048】

そして、このようにバスリセットが発生すると、ノードIDなどのトポロジ情報がクリアされる。そして、その後、トポロジ情報が自動的に再設定される。即ち、バスリセット後、ツリー識別、自己識別が行われる。その後、アイソクロナスリソースマネージャ、サイクルマスタ、バスマネージャ等の管理ノードが決定される。そして、通常の packets 転送が開始される。

【0049】

このように IEEE 1394 では、バスリセット後にトポロジ情報が自動的に再設定されるため、電子機器のケーブルを自由に抜き差しできるようになり、いわゆるホットプラグを実現できる。

【0050】

なお、トランザクションの途中でバスリセットが発生した場合には、そのトランザクションは中止される。そして、中止されたトランザクションを発行した要求ノードは、トポロジ情報を再設定された後に、要求 packets を再度転送する。また、応答ノードは、バスリセットにより中止されたトランザクションの応答 packets を要求ノードに返送してはならない。

【0051】

2. 全体構成

次に、本実施形態のデータ転送制御装置の全体構成の例について図4を用いて説明する。

【0052】

図4において、PHY インターフェース 10 は、PHY デバイス（物理層のデバイス）とのインターフェースを行う回路である。

【0053】

リンクコア 20（リンク手段）は、リンク層のプロトコルやトランザクション層のプロトコルの一部をハードウェアにより実現する回路であり、ノード間での packets 転送のための各種サービスを提供する。レジスタ 22 は、これらのプロトコルを実現したリンクコア 20 を制御するためのレジスタである。

【0054】

FIFO (ATF) 30、FIFO (ITF) 32、FIFO (RF) 34は、各々、非同期送信用、アイソクロナス送信用、受信用のFIFOであり、例えばレジスタや半導体メモリなどのハードウェアにより構成される。本実施形態では、これらのFIFO 30、32、34の段数は非常に少ない。例えば1つのFIFOの段数は、好ましくは3段以下であり、更に好ましくは2段以下となる。

【0055】

DMAC 40 (読み出し手段)、DMAC 42 (読み出し手段)、DMAC 44 (書き込み手段)は、各々、ATF用、ITF用、RF用のDMAコントローラである。これらのDMAC 40、42、44を用いることで、CPU 66に介入されることなく、RAM 80とリンクコア20との間でのデータ転送が可能になる。なお、レジスタ46は、DMAC 40、42、44などを制御するレジスタである。

【0056】

ポートインターフェース50は、アプリケーション層のデバイス (例えばプリンタの印字処理を行うデバイス) とのインターフェースを行う回路である。本実施形態では、このポートインターフェース50を用いて、例えば8ビットのデータ転送が可能になっている。

【0057】

FIFO (PF) 52は、アプリケーション層のデバイスとの間でのデータ転送のためのFIFOであり、DMAC 54は、PF用のDMAコントローラである。レジスタ56は、ポートインターフェース50やDMAC 54を制御するレジスタである。

【0058】

SBP-2コア84は、SBP-2のプロトコルの一部をハードウェアにより実現する回路である。レジスタ88は、SBP-2コア84を制御するためのレジスタである。DMAC (SBP-2用) 86は、SBP-2コア84用のDMAコントローラである。

【0059】

RAM領域管理回路300は、RAM80の各領域を管理するための回路である。RAM領域管理回路300は、RAM80の各領域がフルになったり、エンプティになった場合に、各種のフル信号、エンプティ信号を用いてDMAC40、42、44、54、86を制御する。

【0060】

CPUインターフェース60は、データ転送制御装置をコントロールするCPU66とのインターフェースを行う回路である。CPUインターフェース60は、アドレスデコーダ62、データ同期化回路63、割り込みコントローラ64を含む。クロック制御回路68は、本実施形態で使用されるクロックを制御するものであり、PHYデバイス（PHYチップ）から送られてくるSCLKや、マスタークロックであるHCLKが入力される。

【0061】

バッファマネージャ70は、RAM80とのインターフェースを管理する回路である。バッファマネージャ70は、バッファマネージャの制御のためのレジスタ72、RAM80へのバス接続を調停する調停回路74、各種の制御信号を生成するシーケンサ76を含む。

【0062】

RAM80は、ランダムアクセス可能なパケット記憶手段として機能するものであり、その機能は例えばSRAM、SDRAM、DRAMなどにより実現される。

【0063】

なおRAM80は、本実施形態のデータ転送制御装置に内蔵させることが特に望ましいが、その一部又は全部を外付けにすることも可能である。

【0064】

図5に、RAM80のメモリマップの一例を示す。図5に示すように本実施形態では、RAM80が、ヘッダ領域（AR2、AR3、AR4、AR6）とデータ領域（AR5、AR7、AR8、AR9）に分離されている。そして、パケットのヘッダ（広義には制御情報）はヘッダ領域に格納され、パケットのデータ（ORB、ストリーム）はデータ領域に格納される。

【0065】

また本実施形態では、図5に示すように、RAM80のデータ領域（AR5、AR7、AR8、AR9）が、ORB領域（AR5、AR7）とストリーム領域（AR8、AR9）に分離されている。

【0066】

更に本実施形態では、RAM80が、受信領域（AR2、AR4、AR5、AR9）と送信領域（AR3、AR6、AR7、AR8）に分離されている。

【0067】

なお、ORB（第1の層用の第1のデータ）は、上述したようにSBP-2用のデータ（コマンド）である。一方、ストリーム（第1の層より上層の第2の層用の第2のデータ）は、アプリケーション層用のデータ（プリンタの印字データ、CD-RWの読み出し・書き込みデータ、スキャナによる取り込み画像データ等）である。

【0068】

また、AR1、AR2、AR3に示すHW（ハードウェア）用ページテーブル領域、HW用受信ヘッダ領域、HW用送信ヘッダ領域は、図4に示すSBP-2コア84（SBP-2をハードウェアにより実現する回路）が、ページテーブルや受信ヘッダや送信ヘッダを書き込んだり読み出したりするための領域である。

【0069】

また、図5においてAR4、AR5、AR8、AR9に示す領域は、いわゆるリングバッファ構造になっている。

【0070】

さて、図4のバス90（或いはバス92、94）は、アプリケーションに接続されるものである（第1のバス）。またバス95（或いはバス96）はデータ転送制御装置をコントロールし、或いはデータをリード・ライトするためのものであり、データ転送制御装置をコントロールするデバイス（例えばCPU）に電氣的に接続される（第2のバス）。またバス100（或いはバス102、104、105、106、107、108、109）は、物理層のデバイス（PHYデバイス）に電氣的に接続されるものである（第3のバス）。また、バス110は、

ランダムアクセス可能な記憶手段であるRAM 80に電氣的に接続されるものである(第4のバス)。またバス99は、SBP-2コア84がハードウェアによりSBP-2を実現するためのヘッダ情報やページテーブル情報をリード・ライトするためのものである(第5のバス)。

【0071】

バッファマネージャ70の調停回路74は、DMAC 40、42、44、CPUインターフェース60、DMAC 86、54からのバスアクセス要求の調停を行う。そして、この調停結果に基づいて、各々、バス105、107、109、96、99、94のいずれかと、RAM 80のバス110との間にデータの経路が確立される(第1、第2、第3、第5のバスのいずれかと第4のバスとの間にデータ経路が確立される)。

【0072】

本実施形態の1つの特徴は、ランダムアクセスが可能でありパケットを格納するRAM 80を設けると共に、互いに分離されるバス90、95、99、100と、これらのバスをRAM 80のバス110に接続するための調停回路74とを設けた点にある。

【0073】

例えば図6に、本実施形態と構成の異なるデータ転送制御装置の例を示す。このデータ転送制御装置では、リンクコア902は、PHYインターフェース900、バス922を介してPHYデバイスと接続される。また、リンクコア902は、FIFO 904、906、908、CPUインターフェース910、バス920を介してCPU 912に接続される。そして、CPU 912は、バス924を介して、CPUにローカルなメモリであるRAM 914に接続される。

【0074】

図6の構成のデータ転送制御装置を用いた場合のデータ転送の手法について図7を用いて説明する。PHYデバイス930を介して他のノードから送られてきた受信パケットは、バス922、データ転送制御装置932、バス920を介してCPU 912が受け取る。そして、CPU 912は、受け取った受信パケットをバス924を介してRAM 914に一旦書き込む。そして、CPU 912は、

RAM 914 に書き込まれた受信パケットをバス 924 を介して読み出し、アプリケーション層が使用できるように加工し、バス 926 を介してアプリケーション層のデバイス 934 に転送する。

【0075】

一方、アプリケーション層のデバイス 934 からのデータを転送する場合には、CPU 912 は、このデータを RAM 914 に書き込む。そして、RAM 914 のデータにヘッダを付加することで IEEE 1394 に準拠したパケットを生成する。そして生成されたパケットは、データ転送制御装置 932、PHY デバイス 930などを介して他のノードに送信される。

【0076】

しかしながら、このような図 7 のデータ転送手法によると、CPU 912 の処理負担が非常に重くなる。従って、ノード間を接続するシリアルバスの転送速度が高速になっても、CPU 912 の処理のオーバーヘッドなどに起因して、システム全体の実転送速度は低くなり、結局、高速なデータ転送を実現できない。

【0077】

これに対して、本実施形態では図 8 に示すように、データ転送制御装置 120、アプリケーション層のデバイス 124 間のバス 90 と、CPU バス 96 と、データ転送制御装置 120、RAM 80 間のバス 110 とが互いに分離されている。従って、CPU バス 96 をデータ転送の制御のみに使用できるようになる。また、バス 90 を占有して、データ転送制御装置 120、アプリケーション層のデバイス 124 間でデータ転送を行うことができるようになる。例えば、データ転送制御装置 120 が組み込まれる電子機器がプリンタである場合には、バス 90 を占有して印字データを転送できるようになる。この結果、CPU 66 の処理負担を軽減でき、システム全体の実転送速度を高めることができる。また CPU 66 として安価なものを採用できると共に、CPU バス 96 として高速なバスを使用する必要性がなくなる。このため、電子機器の低コスト化、小規模化を図れるようになる。

【0078】

3. 本実施形態の特徴

3. 1 バスリセットによりトグルするビット

図9 (A) に示すように、IEEE 1394のトランザクションは、要求ノードが応答ノードに要求パケットを送信し、応答ノードからの応答パケットを要求ノードが受信することで完了する。そして、このようなトランザクションの完了後にバスリセットが発生しても問題はない。

【0079】

一方、トランザクションの途中でバスリセットが発生すると、図9 (B) のC1に示すようにトランザクションは中止される。そして、この場合に応答ノードは、中止されたトランザクションについての応答パケットを、要求ノードに対して返送してはならない。また、トランザクションを完了させるためには、要求ノードは、C2に示すように要求パケットを応答ノードに再度送信する必要がある。

【0080】

しかしながら、プリンタやCD-RWなどの周辺機器に本実施形態のデータ転送制御装置を組み込んだ場合、製品コストの制約から、図4のCPU66としては安価で処理能力が低いCPUが使用されるのが一般的である。従って、CPU66上で動作するファームウェアの処理能力も低い。このため、パケットを受信しても、受信したパケットを直ぐには処理できず、多くの未処理の受信パケットがRAM80に存在するようになる。従って、バスリセットが発生した場合には、これらの未処理のパケットが、バスリセットの前後のいずれに受信したパケットなのかを区別する処理が必要になる。即ち、バスリセットの発生場所を検出する処理が必要になる。そして、前述のようにファームウェアの処理能力は一般的に低いと考えられるため、バスリセットの発生場所を検出する処理は、負荷の低いものであることが望まれる。

【0081】

そこで、本実施形態では図10に示すような手法を採用している。

【0082】

即ち、1つのバスリセット（ノードのトポロジ情報をクリアするリセット）から次のバスリセットまでの間をバスリセットインターバルと定義したとする。例

例えば図10では、バスリセットMからM+1までの間はバスリセットインターバルMとなり、バスリセットM+1からM+2までの間はバスリセットインターバルM+1となる。

【0083】

この場合に本実施形態では、受信したパケットと次に受信したパケットとが異なるリセットインターバルに受信したパケットか否かを区別するためのトグルビットBT（広義には区別情報）を生成する。そして、このトグルビットBTを各パケットに関連づけて、図10のC11に示すようにRAM（パケット記憶手段）に書き込むようにしている。

【0084】

即ち図10において、パケットNとN+1は同一のバスリセットインターバルMに受信されているため、これらのBTは共に0になる。一方、パケットN+1とN+2は異なるバスリセットインターバルMとM+1に受信されているため、パケットN+1のBTは0となり、パケットN+2のBTは1になる。即ち、BTが0から1にトグルする。同様に、パケットN+4とN+5も異なるバスリセットインターバルM+1とM+2に受信されているため、BTが1から0にトグルする。また、パケットN+5とN+6も異なるバスリセットインターバルM+2とM+4に受信されているため、BTが0から1にトグルする。

【0085】

従って、図10のC12、C13、C14から明らかなように、BTがトグルした場所がバスリセットの発生場所（RAM上での境界）に対応するようになる。このため、ファームウェア（処理手段）は、BTがトグルした場所を調べるだけで、バスリセットの発生場所を容易に知ることができるようになる。この結果、ファームウェアは、最後のバスリセット後に受信したパケットN+6、N+7、N+8については例えば通常の処理を行い、最後のバスリセット前に受信したパケットN～N+5については破棄する等の処理を行うことができるようになる。

【0086】

そして、本実施形態において特に特徴的なのは、図10のC15においてBT

が変化しないことである。即ち、例えば、バスリセットが発生する毎にBTがトグルようにすると、C15ではバスリセットM+4が発生しているため、BTが1から0にトグルようになる。すると、パケットN+5とN+6は異なるバスリセットインターバルで受信したパケットであるにもかかわらず、図10のC14でBTが0から1に変化しなくなる。この結果、パケットN+5とN+6の受信の間にバスリセットが発生したことをファームウェアが検出できなくなる問題が生じる。

【0087】

本実施形態では、連続して受信したパケットが異なるリセットインターバルに受信したパケットであることを条件に、BTが0から1に或いは1から0にトグルようになるため、上記のような問題は生じない。

【0088】

なお、本実施形態では図5で説明したようにRAMをヘッダ領域とデータ（ORB、ストリーム）領域に分離する。また、ヘッダ領域に格納される各ヘッダと、データ領域に格納される各データとを、ヘッダに含ませたデータポインタにより対応させている。そして本実施形態では図11に示すように、上記のトグルビットBT（区別情報）を、ヘッダ領域に書き込まれるヘッダの中に含ませている。このようにすれば、ファームウェアはヘッダ領域のヘッダをまとめて読み出し、これらのヘッダに含まれるBTを調べるだけで、バスリセットの発生場所を容易に検出できるようになる。この結果、ファームウェアの処理負担を更に軽減できるようになる。

【0089】

なお、図10では、区別情報が1ビットのデータである場合について説明したが、区別情報を2ビット以上のデータにしてもよい。例えば、図10のC12、C13、C14において、0から1、1から0、0から1というように変化させる代わりに、1、2、3というようにその値をインクリメントさせるようにしてもよい。

【0090】

3. 2 バスリセットポインタ

さて、本実施形態では、バスリセットの発生場所を効率的に検出するために、図12に示すようなバスリセットポインタレジスタ（第1のポインタ記憶手段）BPRを設けている。

【0091】

ここで、バスリセットポインタレジスタBPRに記憶されるバスリセットポインタBPは、バスリセット発生前の受信パケットN~N+2とバスリセット発生後の受信パケットN+3~N+6との、RAMにおける境界RB1を特定するポインタである。より具体的には、ポインタBPは、バスリセット発生の直前に受信したパケットN+2の次のパケットN+3の先頭アドレスを指している。

【0092】

更に、本実施形態では、図12に示すように、処理済みポインタレジスタUPR（第2のポインタ記憶手段）や受信済みポインタレジスタPPR（第3のポインタ記憶手段）も設けている。

【0093】

ここで、レジスタUPRに記憶される処理済みポインタUPは、処理済み（使用済み）パケットN-1と未処理（未使用）のパケットNとの、RAMにおける境界RBP2を特定するポインタである。より具体的には、ポインタUPは、処理済みパケットN-1の次のパケットNの先頭アドレスを指している。

【0094】

また、レジスタPPRに記憶される受信済みポインタPPは、最新（post）の受信済みパケットN+6と、未受信のパケットN+7（次に受信するであろうパケット）との、RAMにおける境界RBP3を特定するポインタである。より具体的には、ポインタPPは、最新の受信済みパケットN+6の次に来るべき未受信のパケットN+7の先頭アドレスを指している。

【0095】

上記のレジスタBPRを設けることで、ファームウェアは、バスリセット前の受信パケットとバスリセット後の受信パケットとを、容易に区別できるようになる。また、レジスタUPRやPPRを設けることで、ファームウェアは、どのパケットが未処理のパケットかを、容易に判断できるようになる（図12ではパケ

ットN~N+6が未処理になる)。

【0096】

特に、ポインタBPは、バスリセット直後の受信パケットN+3の先頭アドレスを指している。従って、ファームウェアは、レジスタBPRからポインタBPを読み出すだけで、バスリセット後のパケットに対する処理を開始できるようになる。また、ポインタUPは未処理のパケットNの先頭アドレスを指している。

従って、ファームウェアは、レジスタUPRからポインタUPを読み出すだけで、未処理のパケットに対する処理を開始できるようになる。

【0097】

さて、バスリセット前の受信パケットとバスリセット後の受信パケットとを区別する手法として、いわゆるバスリセットパケットを利用する手法が考えられる。このようなバスリセットパケットを用いれば、図13(A)に示すように、バスリセットパケットの前に格納されているパケットN~N+2は、バスリセット前の受信パケットであると判断でき、バスリセットパケットの後に格納されているパケットN+3~N+6は、バスリセット後の受信パケットであると判断できる。

【0098】

しかしながら、この手法では、図13(B)のフローチャートのステップS1、S2に示すように、ファームウェアは、バスリセットパケットが読み出されるまで未処理の受信パケットをRAMから順次読み出さなければならない。従って、ファームウェアの処理負担が重くなるという問題があり、特に、RAMに多くの未処理のパケットが積まれた場合には、上記問題は更に深刻になる。

【0099】

これに対して、バスリセットポインタBPを用いる本実施形態では、図13(C)のフローチャートのステップS3に示すように、ファームウェアはレジスタBPRからポインタBPを読み出すだけでよい。従って、図13(B)に比べてファームウェアの処理負担を格段に軽減できる。

【0100】

さて、本実施形態では、図5に示すようにRAMをヘッダ領域とデータ領域に

分離している。このため、図12のバスリセットポインタレジスタBRとして、図14に示すようにバスリセットヘッダポインタレジスタBHPR（第4のポインタ記憶手段）とバスリセットORBポインタレジスタBOPR（第5のポインタ記憶手段）を設けている。

【0101】

また、処理済みポインタレジスタUPRとして、処理済みヘッダポインタレジスタUHPRと処理済みORBポインタレジスタUOPRを設けている。また、受信済みポインタレジスタPPRとして、受信済みヘッダポインタレジスタPHPRと受信済みORBポインタレジスタPOPRを設けている。

【0102】

ここで、レジスタBHPR、UHPR、PHPRが記憶するポインタBHP（第4のポインタ情報）、UHP、PHPは、各々、RAMのヘッダ領域での境界RB11、RB21、RB31を特定するためのポインタである。

【0103】

また、レジスタBOPR、UOPR、POPRが記憶するポインタBOP（第5のポインタ情報）、UOP、POPは、各々、RAMのORB（第1のデータ）領域での境界RB12、RB22、RB32を特定するためのポインタである。

【0104】

さて、図14に示すように、ORB領域でのバスリセットの境界RB12を指すポインタBOPを用いることで、次のような利点を得ることができる。

【0105】

即ち、ヘッダ領域での境界RB11を指すポインタBHPしか用いない手法では、ORB領域での境界RB12を特定するために、ファームウェアは図15（A）のフローチャートに示すような処理を行う必要がある。

【0106】

まず、ポインタUOPのアドレスを記憶する（ステップS10）。次に、ポインタBHPの前にヘッダがあるか否かを判断し（ステップS11）、ある場合にはそのヘッダを読み出す（ステップS12）。例えば図14では、ヘッダNが読

み出される。

【0107】

次に、読み出されたヘッダが、ORB領域にデータを持つパケット（ORBパケット）のヘッダか否かを判断する（ステップS13）。図14ではヘッダN、N+1はORBパケットのヘッダではないため、ステップS14には移行せず、ステップS11、S12に戻る。一方、ヘッダN+2、N+3はORBポインタであるので、ステップS14に移行し、ヘッダに含まれるデータポインタ及びデータ長に基づき、ORBポインタのアドレス（境界RB02、RB12）を算出し記憶する。そして、次のヘッダN+4は、ORBポインタではないため、ステップS11に戻る。すると、ステップS11で、ポインタBHPの前にヘッダが無いと判断されるため、ステップS15に移行し、直前に記憶したORBポインタのアドレス（RB12）を、バスリセットによるORB領域の境界であると判断する。

【0108】

以上のように、ポインタBHPのみを用いる手法では、ファームウェアは、図15（A）に示すような負荷の重い処理を行わなければならない。特に、ファームウェアの処理能力が低く、RAMに多くのヘッダが積まれているような状況では、事態は更に深刻になる。

【0109】

これに対して、ポインタBOPを用いれば、図15（B）のフローチャートに示すように、ファームウェアは、レジスタBOPRからポインタBOPを読み出すだけで、境界RB12を特定できる。従って、ポインタBHPしか用いない手法に比べて、ファームウェアの処理負担を格段に軽減できる。

【0110】

さて、SBP-2においては、各ノード（イニシエータ、ターゲット）は、通常の16ビットのノードIDの他に64ビットのEUI-64と呼ばれるIDを持っている。そして、ノードIDはバスリセットによりリセットされ、バスリセット後には全く異なるIDになってしまう可能性が常にあるのに対し、EUI64は、各ノードにユニークなIDであり、バスリセット後も変化しない。従って、

バスリセット後に、EUI-64と新たなノードIDとを対応づける処理が必要になり、各ノード間では、この対応づけの処理のための多数のパケットが送受信される。このため、バスリセット後には、短時間に多くのパケットがRAMに積まれることになる。そして、これらの積まれるパケットの数は、バスに接続されるノードが増えるにつれて多くなる。

【0111】

この場合に、各ノードのファームウェア（トランザクション層）が、バスリセット前の受信パケットに対する処理を優先的に行うと、そのノードの処理がストールしてしまう可能性があることが判明した。そして、1つのノードの処理がストールしてしまうと、その影響は他のノードにも及ぶ。

【0112】

そこで本実施形態では、バスリセットが発生した場合に、ファームウェア（処理手段）が、バスリセット後の受信パケットを優先的に処理するようにしている。

【0113】

即ち、図16のフローチャートに示すように、バスリセットが発生したと判断されると（ステップS20）、ファームウェアは、レジスタBHPR、BOPRからポインタBHP、BOPを読み出す（ステップS21）。そして、バスリセット後の受信パケット（ノードIDとEUI-64との対応づけのためパケット）を優先的に処理する（ステップS22）。即ち、本実施形態では、前述のように、ポインタBHP、BOPが用意されるため、これらのポインタをレジスタBHPR、BOPRから読み出すという簡易な処理で、バスリセット後の受信パケットを特定できる。従って、このようにバスリセット後の受信パケットを優先的に処理するようにしても、ファームウェアの処理負担はそれほど増加しない。

【0114】

次に、ファームウェアは、レジスタUHPR、UOPRからポインタUHP、UOPを読み出し（ステップS23）、バスリセット前の受信パケットの処理を行う（ステップS24）。即ち、パケットを破棄したり、バスリセットにより中止されたトランザクションがあるか否かを判断する等の処理を行う。

【0115】

3. 3 バスリセット送信中止ステータス

図17 (A) に示すように、通常の送信処理は、ファームウェアが送信開始コマンドを発行することで（図4のレジスタ46に送信開始コマンドを書き込むことで）、開始される。そして、送信開始コマンドが発行されると、バスのアービトレーションが行われ、アービトレーションに勝つと、バス上での実際のパケット転送が開始する。そして、相手ノードからACK（アクノリジメント）が返ってくると、送信完了のステータスがファームウェアに伝えられる。

【0116】

そして、図17 (B) に示すように、送信開始コマンドの発行前にバスリセットが発生していた場合には、図17 (A) と同様に通常の送信処理が行われる。一方、図17 (C) に示すように、送信開始コマンドの発行後にバスリセットが発生すると、送信は中止され、相手ノードからのACKは返って来ず、送信完了のステータスはファームウェアには伝えられない。

【0117】

ところが、ファームウェアによる送信開始コマンドの発行（レジスタへの書き込み）の少し前にバスリセットが起きたような場合には、図17 (B)、(C) のどちらのケースになったのかを、ファームウェアが判断できなくなる。

【0118】

即ち図18 (A) に示すようにファームウェアは、送信開始コマンドを発行した後（ステップS30）、バスリセットが発生したか否かを判断する（ステップS31）。そして、バスリセットが発生していない場合には、送信完了が返ってくるのを待つことになる（ステップS32）。これは、図17 (B) のケースである。一方、バスリセットが発生した場合には、送信完了を待つことなく、送信をキャンセルする（ステップS33）。これは図17 (C) のケースである。

【0119】

そして、図18 (A) のC20に示すように、送信開始コマンドの発行よりも少し前にバスリセットが発生した場合には、ファームウェアの処理がステップS31、S32でループしてしまい、処理がストールしてしまう。即ち、バスリセ

ットの発生を検出できないため、送信はキャンセルされず（ステップS33に移行しない）、送信完了も返って来ないため、ステップS31、S2で処理がループする。

【0120】

そこで本実施形態では図17（D）に示すように、バスリセットにより送信が中止されたことを示すステータスをファームウェアに伝えるようにしている。より具体的には、バスリセットが発生し、データ転送制御装置のハードウェアが送信を中止する処理を行った場合に、バスリセット送信中止ステータスが図4のレジスタ46に書き込まれる。このようにすればファームウェアは、図18（B）のステップS41に示すように、バスリセットにより送信が中止されたか否かを判断できる。そして、バスリセットにより送信が中止されたと判断した場合には、送信完了を待つことなく送信をキャンセルするようにする（ステップS44）。これにより、ファームウェアの処理がストールしてしまう事態を回避できるようになる。

【0121】

4. 詳細例

4.1 受信側の詳細な構成

次に受信側の詳細な構成について説明する。図19に、リンクコア20（リンク手段）、FIFO34、DMAC44（書き込み手段）の詳細な構成の一例を示す。

【0122】

リンクコア20は、バス監視回路130、直列・並列変換回路132、パケット整形回路160を含む。

【0123】

ここで、バス監視回路130は、PHYインターフェース10を介してPHYデバイスに接続される8ビット幅のデータバスD、2ビット幅のコントロールバスCTLを監視する回路である。

【0124】

直列・並列変換回路132は、データバスDのデータを32ビットのデータに

変換する回路である。

【0125】

パケット整形回路160は、各ノードから転送されてきたパケットを上層が使用できるように整形する回路である。例えば図20(A)に、IEEE1394規格の、非同期でブロックデータを有するパケットのフォーマットを示す。一方、図20(B)に、RAM80のヘッダ領域に格納される、非同期受信でブロックデータを有するパケットのヘッダ部分のフォーマットを示す。このように本実施形態では、図20(A)に示すフォーマットのパケットを、ファームウェアなどの上層が使用できるように、図20(B)に示すフォーマットのパケットに整形している。

【0126】

パケット整形回路160は、パケット診断回路142、シーケンサ167、バッファ168、セクタ170を含み、パケット診断回路142は、TAG生成回路162、ステータス生成回路164、エラーチェック回路166を含む。

【0127】

ここでパケット診断回路142は、パケットの診断を行う回路である。TAG生成回路162は、パケットを書き込む領域を区別するための情報であるTAGを生成する回路であり、ステータス生成回路164は、パケットに付加する各種のステータスを生成する回路である。また、エラーチェック回路166は、パケットに含まれるパリティやCRCなどのエラーチェック情報をチェックしてエラーを検出する回路である。

【0128】

シーケンサ167は各種の制御信号を生成するものである。バッファ168、セクタ170は、直列・並列変換回路132からのDI、パケット診断回路142からのステータス、DMAC44からのデータポインタDPのいずれかを、パケット診断回路142からの信号SELにより選択するためのものである。

【0129】

FIFO34は、リンコア20からの出力データであるRDの位相と、RAM80への書き込みデータであるWDATAの位相とを調整するためのバッファと

して機能するものであり、F I F O 状態判断回路 35 を含む。F I F O 状態判断回路 35 は、F I F O 34 が空になると、信号 E M P T Y をアクティブにし、F I F O 34 がフルになると、信号 F U L L をアクティブにする。

【0130】

D M A C 44 は、パケット分離回路 180、アクセス要求実行回路 190、アクセス要求発生回路 192 を含む。

【0131】

ここでパケット分離回路 180 は、パケット整形回路 160 により整形されたパケットを T A G (D T A G) に基づいてデータ、ヘッダ等に分離して、R A M の各領域 (図 5 参照) に書き込む処理を行う。

【0132】

アクセス要求実行回路 190 は、リンクコア 20 からのアクセス要求を実行するための回路である。アクセス要求実行回路 190 は、F I F O 状態判断回路 35 からの F U L L がアクティブになると、F F U L L をアクティブにする。パケット整形回路 160 内のシーケンサ 167 は、F F U L L がアクティブでないことを条件に、R D (R x D a t a) のストロブ信号である R D S をアクティブにする。

【0133】

なお R F A I L は、受信における失敗を、シーケンサ 167 がアクセス要求実行回路 190 に対して知らせるための信号である。

【0134】

アクセス要求発生回路 192 は、R A M 80 へのアクセス要求を発生するための回路である。アクセス要求発生回路 192 は、バッファマネージャ 70 からの書き込みアクノリジメントである W A C K や F I F O 状態判断回路 35 からの E M P T Y などを受け、書き込み要求である W R E Q をバッファマネージャ 70 に出力する。

【0135】

さて、図 19 に示すように、パケット分離回路 180 は、T A G 判別回路 182、アドレス発生回路 188 を含み、アドレス発生回路 188 はポインタ更新回

路 184 を含む。

【0136】

ここで TAG 判別回路 182 は、TAG 生成回路 162 により生成された TAG (DTAG) を判別し、FIFO 34 の出力 WDATA の書き込み領域を決める。

【0137】

そして、アドレス発生回路 188 が含むポインタ更新回路 184 が、この決められた領域において、ポインタ（データポインタ、ヘッダポインタ）を順次更新（インクリメント、デクリメント）する。そして、アドレス発生回路 188 は、この順次更新されるポインタが指すアドレスを発生して、WADR としてバッファマネージャ 70 に出力する。また、アドレス発生回路 188 は、データポインタ DP（受信 ORB 領域のデータポインタ、受信ストリーム領域のデータポインタ等）をパケット整形回路 160 に出力する。パケット整形回路 160 は、このデータポインタをパケットのヘッダに埋め込む（図 20（B）の C30 参照）。このようにすることで、ヘッダ領域に格納されるヘッダとデータ領域に格納されるデータとを対応づけることが可能になる（図 11 参照）。

【0138】

図 21 に、本実施形態で使用される TAG (DTAG) の例を示す。図 21 において、例えば TAG が (0001)、(0010) であった場合には、受信パケットのヘッダ (FIFO 34 の出力 WDATA) が図 5 の受信ヘッダ領域へ書き込まれる。また、TAG が (0100) であった場合には、受信パケットのデータが受信 ORB 領域に書き込まれ、TAG が (0101) であった場合には、受信パケットのデータが受信ストリーム領域に書き込まれることになる。

【0139】

また、TAG が (1001)、(1010) であった場合には、受信パケットのヘッダが HW（ハードウェア）用受信ヘッダ領域に書き込まれる。また、TAG が (1100) であった場合には、受信パケットのデータが HW 用受信 ORB 領域に書き込まれ、TAG が (1101) であった場合には、受信パケットのデータが HW 用受信ストリーム領域に書き込まれることになる。なお、ここで HW

(ハードウェア) 用とは、図 4 の S B P-2 コア 84 用という意味である。

【0140】

4. 2 B T 生成回路

ステータス生成回路 164 は、B T 生成回路 165 を含む。この B T 生成回路 165 は、図 10 で説明したトグルビット B T を生成する。生成された B T は、図 20 (B) の C 31 に示すように、整形後のパケットのヘッダに埋め込まれる。

【0141】

図 22 (A)、(B) に、B T 生成回路 165 の状態遷移図を示す。

【0142】

図 22 (A) において、R E C E I V E D は B T 生成回路 165 の内部信号であり、B R I P はバスリセット中であることを示す信号である。この B R I P は、図 19 に示すようにバス監視回路 130 が生成する。即ちバス監視回路 130 は、データバス D を介して P H Y デバイスからのステータス情報を受け取り、このステータス情報に基づいて、バスリセットがなされたか否かを判断する。そして、バス監視回路 130 は、バスリセットがなされたと判断すると、B R I P を H レベルにして、その後、L レベルに戻す。

【0143】

図 22 (A) の状態遷移図に示すように、R E C E I V E D は、パケットを受信したことを条件に L レベルから H レベルに変化し、B R I P が H レベルになったことを条件に H レベルから L レベルに変化する。また図 22 (B) の状態遷移図に示すように、トグルビット B T は、B R I P 及び R E C E I V E D が H レベルになったことを条件に、L レベルから H レベルに或いは H レベルから L レベルにトグルする。

【0144】

図 23 に、上述した各信号のタイミングチャートを示す。図 23 の C 40、C 41、C 42 では、パケットを受信したため、R E C E I V E D が L レベルから H レベルに変化している。また、C 43、C 44、C 45 では、B R I P (バスリセット中信号) が H レベルになったため、R E C E I V E D が H レベルから L

レベルに変化している。

【0145】

C46、C47、C48では、BRIP及びRECEIVEDがHレベルになったため、BTが、LレベルからHレベルに或いはHレベルからLレベルにトグルしている。一方、C49では、RECEIVEDがHレベルではないため、BTは変化しない。即ち、バスリセットインターバルM+2ではパケットを受信していないため、バスリセットが生じて（BRIPがHレベルになっても）、BTは変化しない。このようにすることで、図10で説明したように、連続して受信したパケットが異なるバスリセットインターバルのパケットである場合に变化するようなトグルビットBTを生成できるようになる。

【0146】

4. 3 ポインタレジスタ

次に、図24を用いて、図14の各種ポインタレジスタの詳細について説明する。

【0147】

レジスタ310、314、318は、各々、受信済みヘッダポインタ、受信済みORBポインタ、受信済みストリームポインタを記憶するレジスタである（図14参照）。これらのレジスタ310、314、318は、アドレス発生回路188から、各々、WHADR（ヘッダ領域でのアドレス）、WOADR（ORB領域でのアドレス）、WSADR（ストリーム領域でのアドレス）を受ける。また、レジスタ310、314、318は、リンクコア20からの受信完了信号RXCOMPを受ける。そして、レジスタ310、314、318は、このRXCOMPがアクティブになるタイミングで、アドレス発生回路188からのWHADR、WOADR、WSADRを取り込み、記憶する。このようにすることで、図14の境界RB31、RB32等のアドレスを記憶できるようになる。

【0148】

またレジスタ312、316は、各々、バスリセットヘッダポインタ、バスリセットORBポインタを記憶するレジスタである（図14参照）。これらのレジスタ312、316は、リンクコア20からのバスリセット中信号BRIPを受

ける。そして、レジスタ 312、316 は、この BRIP がアクティブになるタイミングで、レジスタ 310、314 に記憶されているアドレスを取り込み、記憶する。このようにすることで、図 14 の境界 RB11、RB12 のアドレスを記憶できるようになる。

【0149】

レジスタ 320、322、324 は、各々、処理済みヘッダポインタ、処理済み ORB ポインタ、処理済みストリームポインタを記憶するレジスタである（図 14 参照）。

【0150】

スタート・エンドアドレスレジスタ 326 は、図 5 に示す各領域のスタートアドレスやエンドアドレスを記憶する。そして、アドレス発生回路 188、332 は、レジスタ 326 からのスタートアドレス、エンドアドレスに基づいてアドレスの発生を制御する。より具体的には、スタートアドレスを開始点として順次ポインタを更新する。そして、ポインタがエンドアドレスに到達した場合に、ポインタをスタートアドレスに戻すなどの制御を行う（リングバッファ構造の場合）。

【0151】

RAM 領域管理回路 300 は、受信ヘッダ領域管理回路 302、受信 ORB 領域管理回路 304、受信ストリーム領域管理回路 306 を含む。

【0152】

そして、受信ヘッダ領域管理回路 302 は、レジスタ 310 からの受信済みヘッダポインタやレジスタ 320 からの処理済みヘッダポインタを受け、受信ヘッダ領域がフルであることを知らせる信号 HDRFULL をアクセス要求発生回路 192 に出力する。

【0153】

また、受信 ORB 領域管理回路 304 は、レジスタ 314 からの受信済み ORB ポインタやレジスタ 322 からの処理済み ORB ポインタを受け、受信 ORB 領域がフルであることを知らせる信号 ORBFULL をアクセス要求発生回路 192 に出力する。

【0154】

また、受信ストリーム領域管理回路306は、レジスタ318からの受信済みストリームポインタやレジスタ324からの処理済みストリームポインタを受け、受信ストリーム領域がフルであることを知らせる信号STRMFULLをアクセス要求発生回路192に出力する。また、受信ストリーム領域がエンプティであることを知らせる信号STRMEMPTYをアクセス要求発生回路334に出力する。

【0155】

アクセス要求発生回路192、334は、これらのフル信号、エンプティ信号を受けて、書き込み要求WREQ、読み出し要求RREQをバッファマネージャ70に出力するか否かを決めることになる。

【0156】

4.4 バスリセット送信中止ステータス

次に、図25、図26を用いて、バスリセット送信中止ステータスの詳細について説明する。

【0157】

図25において、ファームウェアは送信開始コマンドを送信開始設定レジスタ340に書き込む。するとSTART生成回路342が、図26のC60に示すように信号STARTをアクティブにする。するとDMAC40がバッファマネージャ70に対して読み出し要求を出力し、送信が開始される。

【0158】

また、信号STARTを受けたTXPRD生成回路346は、C61に示すように、送信中であることを示す信号TXPRDをアクティブにする。そして、パケット転送が無事に終了し、C62に示すようにリンクコア20が送信完了信号TXCOMPをアクティブにすると、信号TXPRDは非アクティブになる。

【0159】

さて、送信中（TXPRDがアクティブの時）に、C63に示すように信号BRIPがアクティブになると（バスリセットが発生すると）、C64に示すように、図25のTXBRABORT生成回路348が信号TXBRABORTをア

クティブにする。そして、バスリセットにより送信が中止されたことを示すステータスが、送信中止ステータスレジスタ 350 を介してファームウェアに伝えられることになる。

【0160】

一方、図 26 の C65 では、送信中ではないため (TXPRD が非アクティブ)、バスリセットが発生して BRIP がアクティブになっても、TXBRABO T はアクティブにならない。

【0161】

このように本実施形態では、送信中にバスリセットが発生して送信が中止になった時にのみ、バスリセット送信中止ステータスがファームウェアに伝えられるようになる。

【0162】

5. 電子機器

次に、本実施形態のデータ転送制御装置を含む電子機器の例について説明する。

【0163】

例えば図 27 (A) に電子機器の 1 つであるプリンタの内部ブロック図を示し、図 28 (A) にその外観図を示す。CPU (マイクロコンピュータ) 510 はシステム全体の制御などを行う。操作部 511 はプリンタをユーザが操作するためのものである。ROM 516 には、制御プログラム、フォントなどが格納され、RAM 518 は CPU 510 のワーク領域として機能する。表示パネル 519 はプリンタの動作状態をユーザに知らせるためのものである。

【0164】

PHY デバイス 502、データ転送制御装置 500 を介して、パーソナルコンピュータなどの他のノードから送られてきた印字データは、バス 504 を介して印字処理部 512 に直接送られる。そして、印字データは、印字処理部 512 にて所与の処理が施され、プリントヘッダなどからなる印字部 (データを出力するための装置) 514 により紙に印字されて出力される。

【0165】

図 27 (B) に電子機器の 1 つであるスキャナの内部ブロック図を示し、図 28 (B) にその外観図を示す。CPU 520 はシステム全体の制御などを行う。操作部 521 はスキャナをユーザが操作するためのものである。ROM 526 には制御プログラムなどが格納され、RAM 528 は CPU 520 のワーク領域として機能する。

【0166】

光源、光電変換器などからなる画像読み取り部（データを取り込むための装置）522 により原稿の画像が読み取られ、読み取られた画像のデータは画像処理部 524 により処理される。そして、処理後の画像データがバス 505 を介してデータ転送制御装置 500 に直接送られる。データ転送制御装置 500 は、この画像データにヘッダなどを付加することでパケットを生成し、PHY デバイス 502 を介してパーソナルコンピュータなどの他のノードに送信する。

【0167】

図 27 (C) に電子機器の 1 つである CD-RW ドライブの内部ブロック図を示し、図 28 (C) にその外観図を示す。CPU 530 はシステム全体の制御などを行う。操作部 531 は CD-RW をユーザが操作するためのものである。ROM 536 には制御プログラムなどが格納され、RAM 538 は CPU 530 のワーク領域として機能する。

【0168】

レーザ、モータ、光学系などからなる読み取り&書き込み部（データを取り込むための装置又はデータを記憶するための装置）533 により CD-RW 532 から読み取られたデータは、信号処理部 534 に入力され、エラー訂正処理などの所与の信号処理が施される。そして、信号処理が施されたデータが、バス 506 を介してデータ転送制御装置 500 に直接送られる。データ転送制御装置 500 は、このデータにヘッダなどを付加することでパケットを生成し、PHY デバイス 502 を介してパーソナルコンピュータなどの他のノードに送信する。

【0169】

一方、PHY デバイス 502、データ転送制御装置 500 を介して、他のノードから送られてきたデータは、バス 506 を介して信号処理部 534 に直接送ら

れる。そして、信号処理部 534 によりこのデータに所与の信号処理が施され、読み取り&書き込み部 533 により CD-RW 532 に記憶される。

【0170】

なお、図 27 (A)、(B)、(C) において、CPU 510、520、530 の他に、データ転送制御装置 500 でのデータ転送制御のための CPU を別に設けるようにしてもよい。

【0171】

また、図 27 (A)、(B)、(C) では RAM 501 (図 4 の RAM 80 に相当) がデータ転送制御装置 500 の外部に設けられているが、RAM 501 をデータ転送制御装置 500 に内蔵させてもよい。

【0172】

本実施形態のデータ転送制御装置を電子機器に用いることで、高速なデータ転送が可能になる。従って、ユーザがパーソナルコンピュータなどによりプリントアウトの指示を行った場合に、少ないタイムラグで印字が完了するようになる。また、スキャナへの画像取り込みの指示の後に、少ないタイムラグで読み取り画像をユーザは見るようになる。また、CD-RW からのデータの読み取りや、CD-RW へのデータの書き込みを高速に行うことができるようになる。更に、例えば 1 つのホストシステムに複数の電子機器を接続して利用したり、複数のホストシステムに複数の電子機器を接続して利用したりすることも容易になる。

【0173】

また本実施形態のデータ転送制御装置を電子機器に用いることで、CPU 上で動作するファームウェアの処理負担が軽減され、安価な CPU や低速のバスを用いることが可能になる。更に、データ転送制御装置の低コスト化、小規模化を図れるため、電子機器の低コスト化、小規模化も図れるようになる。

【0174】

また、バスに新たな電子機器が接続され、バスリセットが発生した場合にも、電子機器間での通常のデータ転送が長時間待たされる事態を防止できるようになる。

【0175】

なお本実施形態のデータ転送制御装置を適用できる電子機器としては、上記以外にも例えば、種々の光ディスクドライブ（CD-ROM、DVD）、光磁気ディスクドライブ（MO）、ハードディスクドライブ、TV、VTR、ビデオカメラ、オーディオ機器、電話機、プロジェクタ、パーソナルコンピュータ、電子手帳、ワードプロセッサなど種々のものを考えることができる。

【0176】

なお、本発明は本実施形態に限定されず、本発明の要旨の範囲内で種々の変形実施が可能である。

【0177】

例えば、本発明のデータ転送制御装置の構成は、図4に示す構成が特に望ましいが、これに限定されるものではない。

【0178】

また、本発明はIEEE1394におけるバスリセットに特に有用だが、これ以外にも、少なくともノードのトポロジー情報をクリアするようリセットであれば適用できる。

【0179】

また本発明のポインタ情報は、パケット記憶手段の各境界を少なくとも特定できるものであればよく、パケットの先頭アドレスには限定されない。

【0180】

またパケット記憶手段の分離（分割）手法も、図5で説明したものに限定されるものではない。

【0181】

また、本発明は、IEEE1394規格でのデータ転送に適用されることが特に望ましいが、これに限定されるものではない。例えばIEEE1394と同様の思想に基づく規格やIEEE1394を発展させた規格におけるデータ転送にも本発明は適用できる。

【図面の簡単な説明】

【図1】

IEEE 1394 の層構造について示す図である。

【図 2】

図 2 (A)、(B) は、トランザクション層やリンク層が提供する各種のサービスについて説明するための図である。

【図 3】

SBP-2 について説明するための図である。

【図 4】

本実施形態のデータ転送制御装置の構成例を示す図である。

【図 5】

RAM (パケット記憶手段) の分離 (分割) 手法について説明するための図である。

【図 6】

比較例の構成について示す図である。

【図 7】

図 6 の構成によるデータ転送の手法について説明するための図である。

【図 8】

本実施形態のデータ転送の手法について説明するための図である。

【図 9】

図 9 (A)、(B) は、バスリセットの発生によるトランザクションの中止について説明するための図である。

【図 10】

トグルビット BT について説明するための図である。

【図 11】

トグルビット BT をヘッダに含ませる手法について説明するための図である。

【図 12】

バスリセットポインタについて説明するための図である。

【図 13】

図 13 (A)、(B)、(C) は、バスリセットパケットを用いる場合とバスリセットポインタを用いる場合のファームウェアの処理について説明するための

図である。

【図 14】

バスリセットORBポインタについて説明するための図である。

【図 15】

図 15 (A)、(B) は、バスリセットヘッダポインタのみを用いる場合とバスリセットORBポインタを用いる場合のファームウェアの処理について説明するためのフローチャートである。

【図 16】

バスリセット後の受信パケットを優先的に処理する手法について説明するためのフローチャートである。

【図 17】

図 17 (A)、(B)、(C)、(D) は、バスリセット送信中止ステータスについて説明するための図である。

【図 18】

図 18 (A)、(B) は、バスリセット送信中止ステータスを用いる場合と用いない場合のファームウェアの処理について説明するためのフローチャートである。

【図 19】

受信側の詳細な構成について示す図である。

【図 20】

図 20 (A) は、IEEE 1394 規格の、非同期でブロックデータを有するパケットのフォーマットであり、図 20 (B) は、RAMに格納される、非同期受信でブロックデータを有するパケットのヘッダ部分のフォーマットである。

【図 21】

TAGについて説明するための図である。

【図 22】

図 22 (A)、(B) は、BT生成回路の状態遷移図である。

【図 23】

BT生成回路の動作について説明するためのタイミング波形図である。

【図 24】

各種ポインタレジスタの詳細について説明するための図である。

【図 25】

バスリセット送信中止ステータスの詳細について説明するための図である。

【図 26】

バスリセット送信中止ステータスの詳細について説明するためのタイミング波形図である。

【図 27】

図 27 (A)、(B)、(C) は、種々の電子機器の内部ブロック図の例である。

【図 28】

図 28 (A)、(B)、(C) は、種々の電子機器の外観図の例である。

【符号の説明】

- 10 PHYインターフェース
- 20 リンクコア
- 22 レジスタ
- 30 FIFO (ATF)
- 32 FIFO (ITF)
- 34 FIFO (RF)
- 40 DMAC (ATF用)
- 42 DMAC (ITF用)
- 44 DMAC (RF用)
- 46 レジスタ
- 50 ポートインターフェース
- 52 FIFO (PF)
- 54 DMAC (PF用)
- 56 レジスタ
- 60 CPUインターフェース
- 62 アドレスデコーダ

63 データ同期化回路

64 割り込みコントローラ

66 CPU

68 クロック制御回路

70 バッファマネージャ

72 レジスタ

74 調停回路

76 シーケンサ

80 RAM (パケット記憶手段)

84 SBP-2コア

86 DMAC (SBP-2用)

90、92、94 バス (第1のバス)

95、96 バス (第2のバス)

99 バス (第5のバス)

100、102、104、105、

106、107、108、109 バス (第3のバス)

110 バス (第4のバス)

120 データ転送制御装置

122 PHYデバイス

124 アプリケーション層のデバイス

130 バス監視回路

132 直列・並列変換回路

142 パケット診断回路

160 パケット整形回路

162 TAG生成回路

164 ステータス生成回路

165 BT生成回路

166 エラーチェック回路

167 シーケンサ

- 168 バッファ
- 170 セクタ
- 180 パケット分離回路
- 182 TAG判別回路
- 184 ポインタ更新回路
- 188 アドレス発生回路

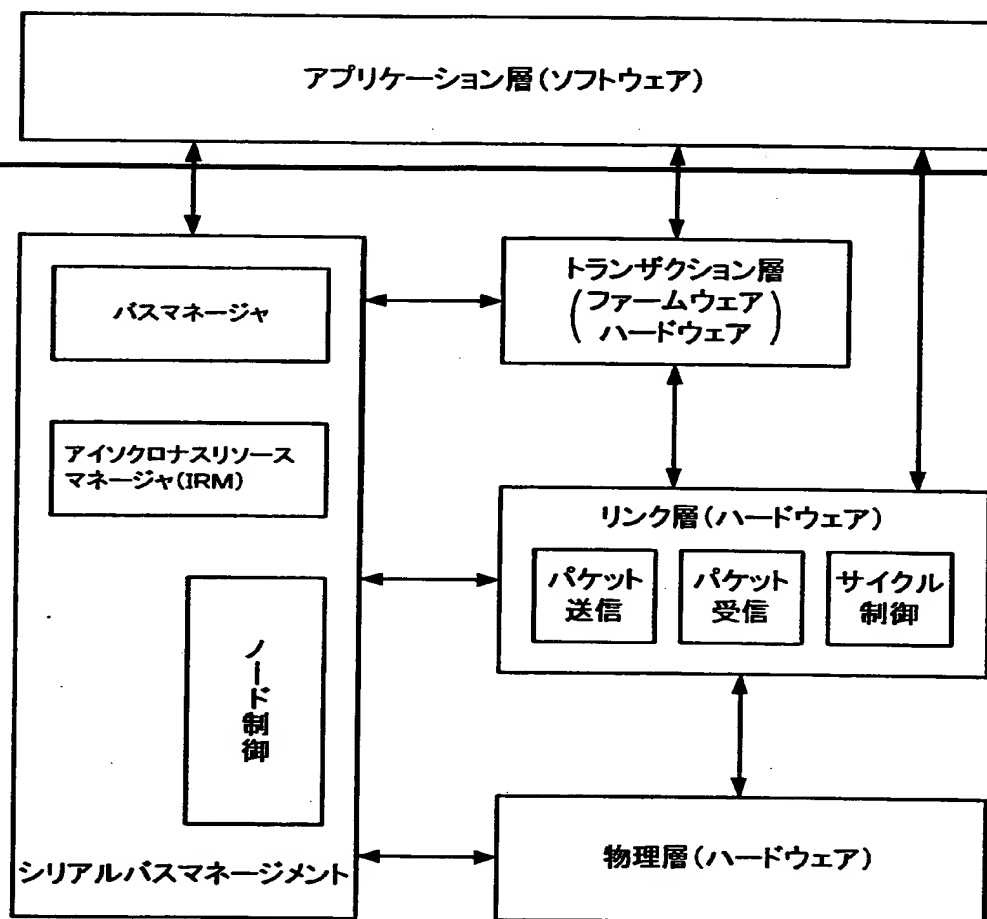
- 190 アクセス要求実行回路
- 192 アドレス要求発生回路
- 300 RAM領域管理回路
- 302 受信ヘッダ領域管理回路
- 304 受信ORB領域管理回路
- 306 受信ストリーム領域管理回路
- 310 受信済みヘッダポインタレジスタ (PHPR)
- 312 バスリセットヘッダポインタレジスタ (BHPR)
- 314 受信済みORBポインタレジスタ (POPR)
- 316 バスリセットORBポインタレジスタ (BOPR)
- 318 受信済みストリームポインタレジスタ (PSPR)
- 320 処理済みヘッダポインタレジスタ (UHPR)
- 322 処理済みORBポインタレジスタ (UOPR)
- 324 処理済みストリームポインタレジスタ (USPR)
- 326 スタート・エンドアドレスレジスタ
- 332 アドレス発生回路
- 334 アクセス要求発生回路
- 340 送信開始設定レジスタ
- 342 START生成回路

- 346 TXPRD生成回路
- 348 TXBRABORT生成回路
- 350 送信中止ステータスレジスタ

【書類名】

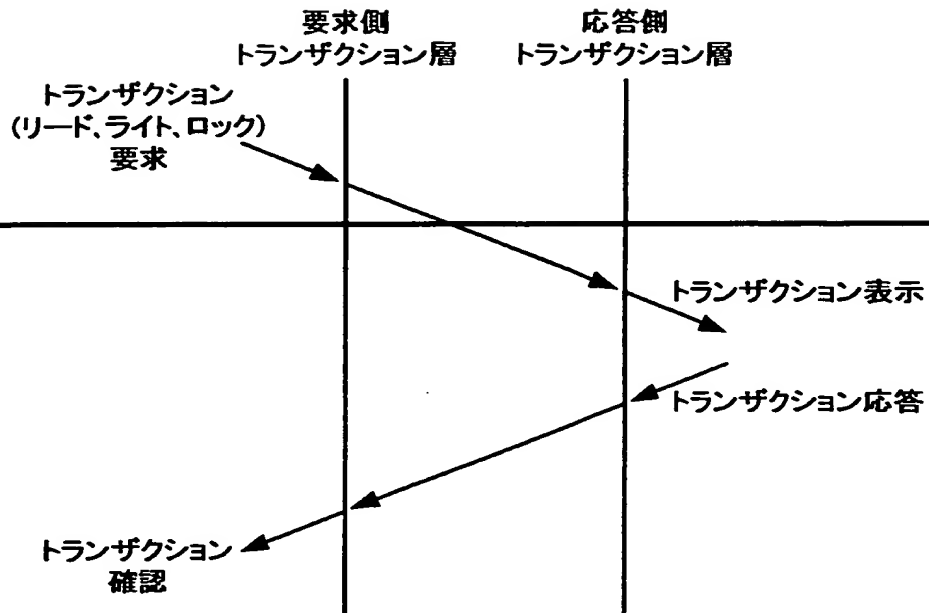
図面

【図 1】

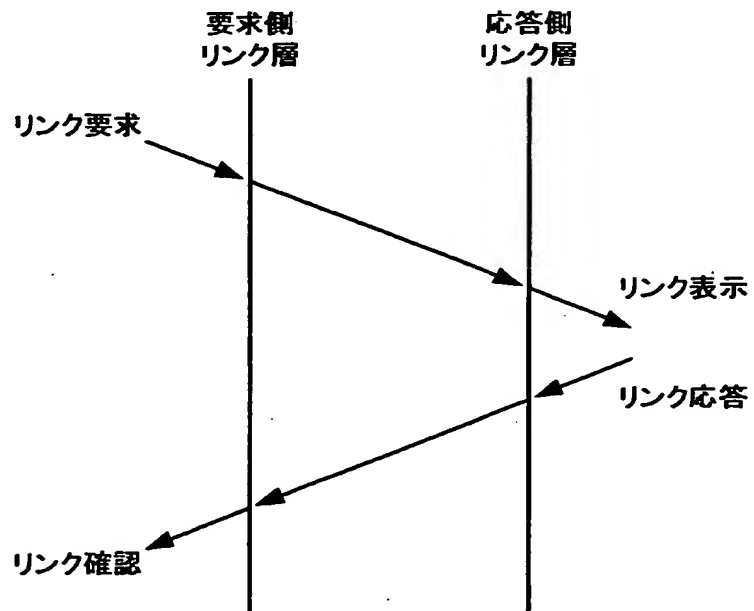


【図 2】

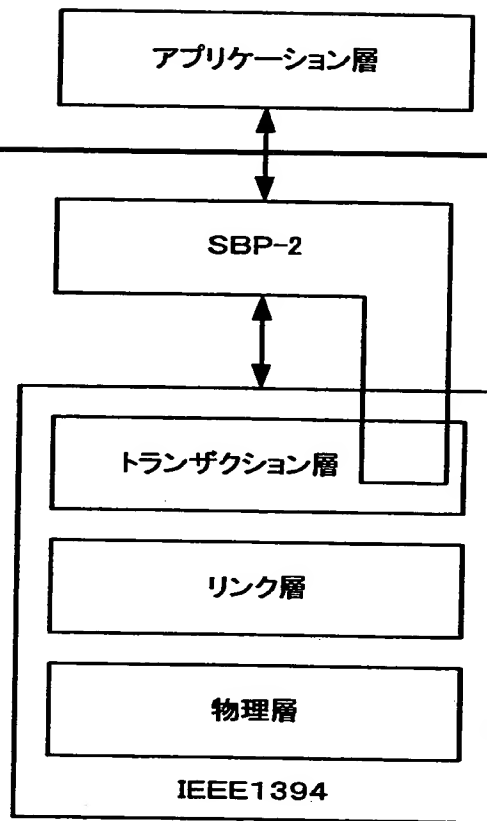
(A)



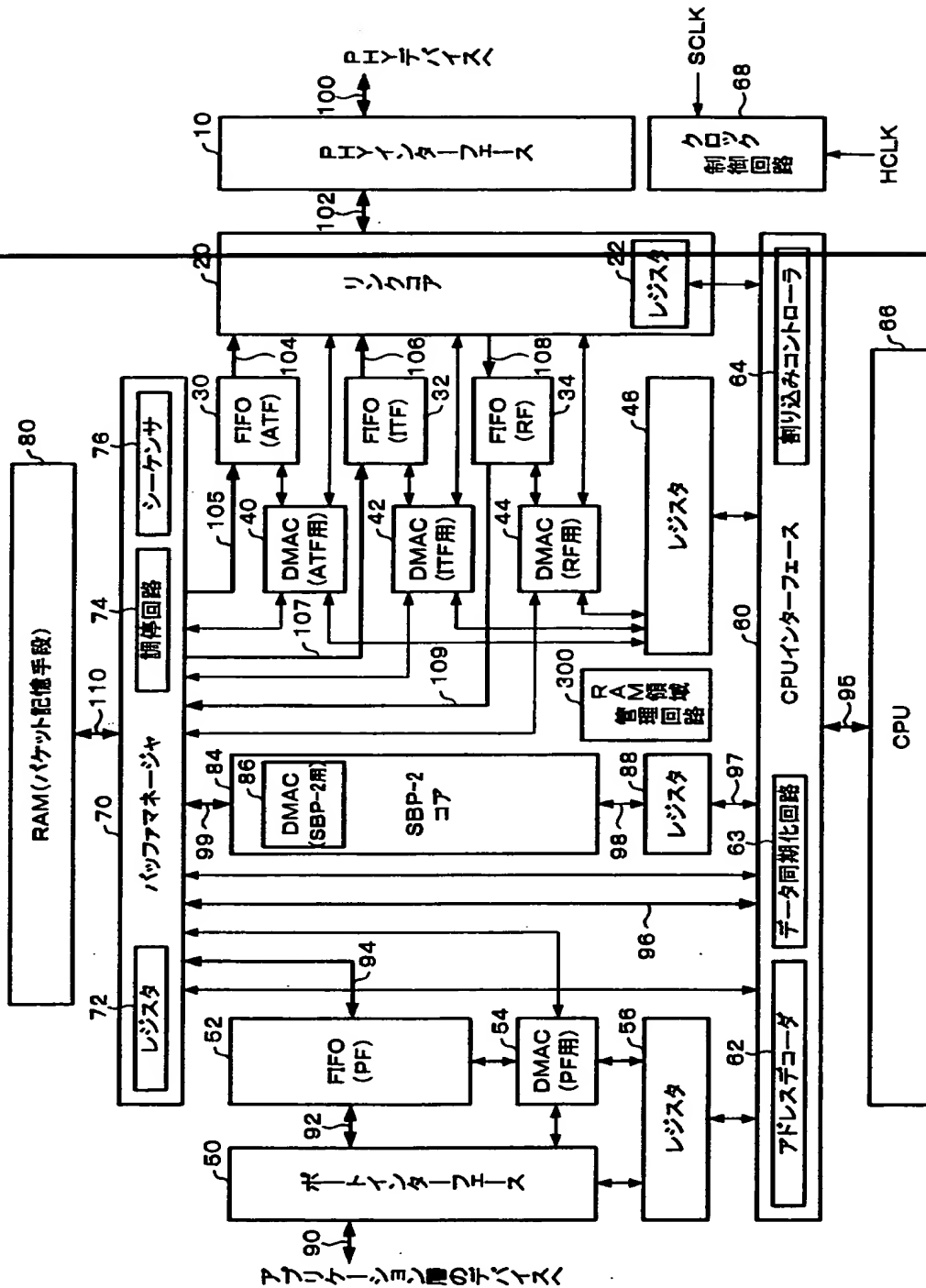
(B)



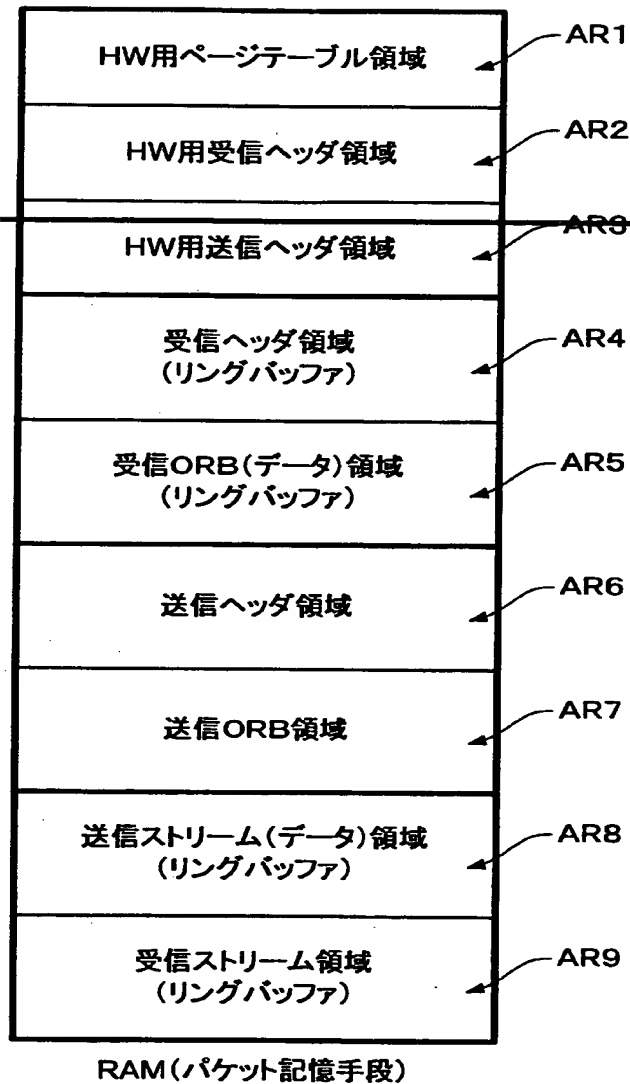
【図 3】



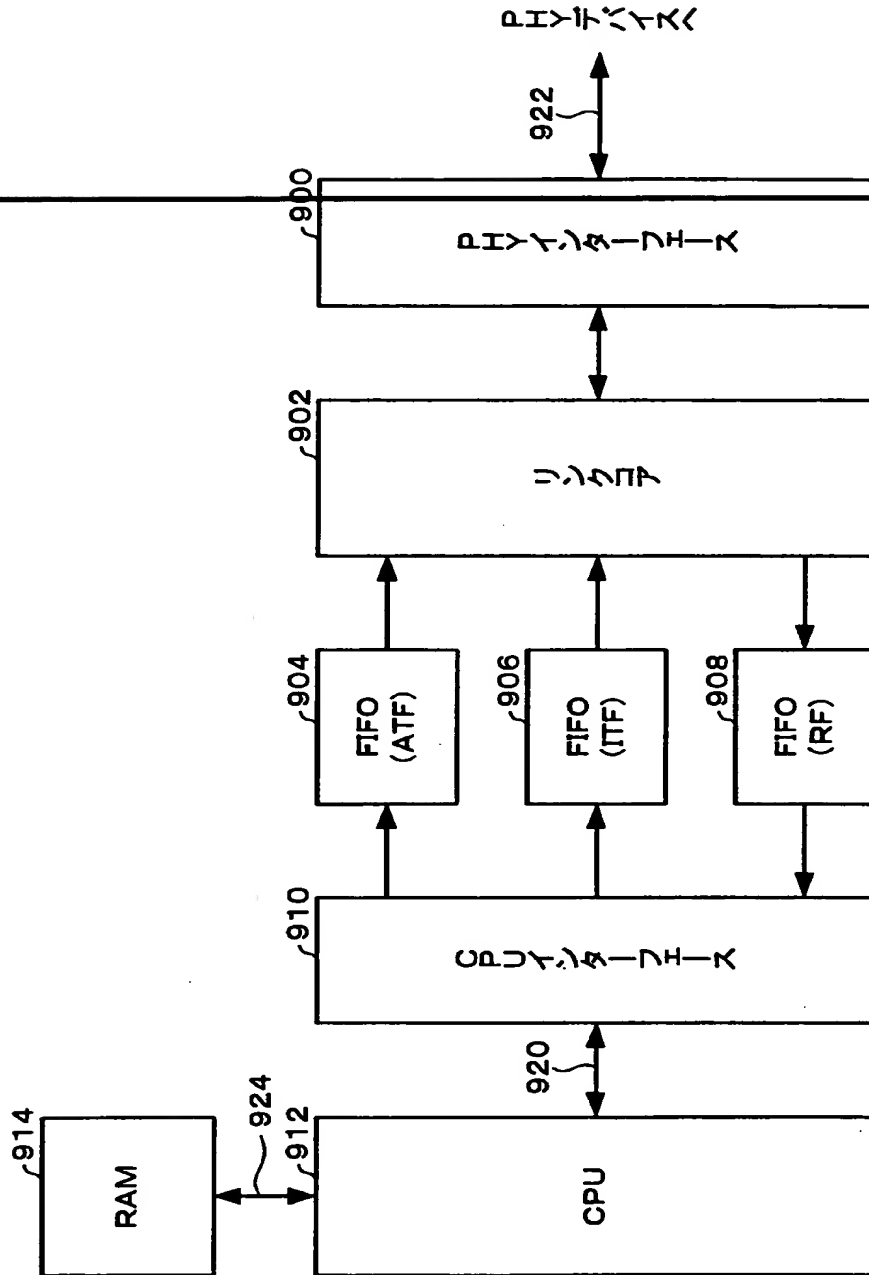
【図 4】



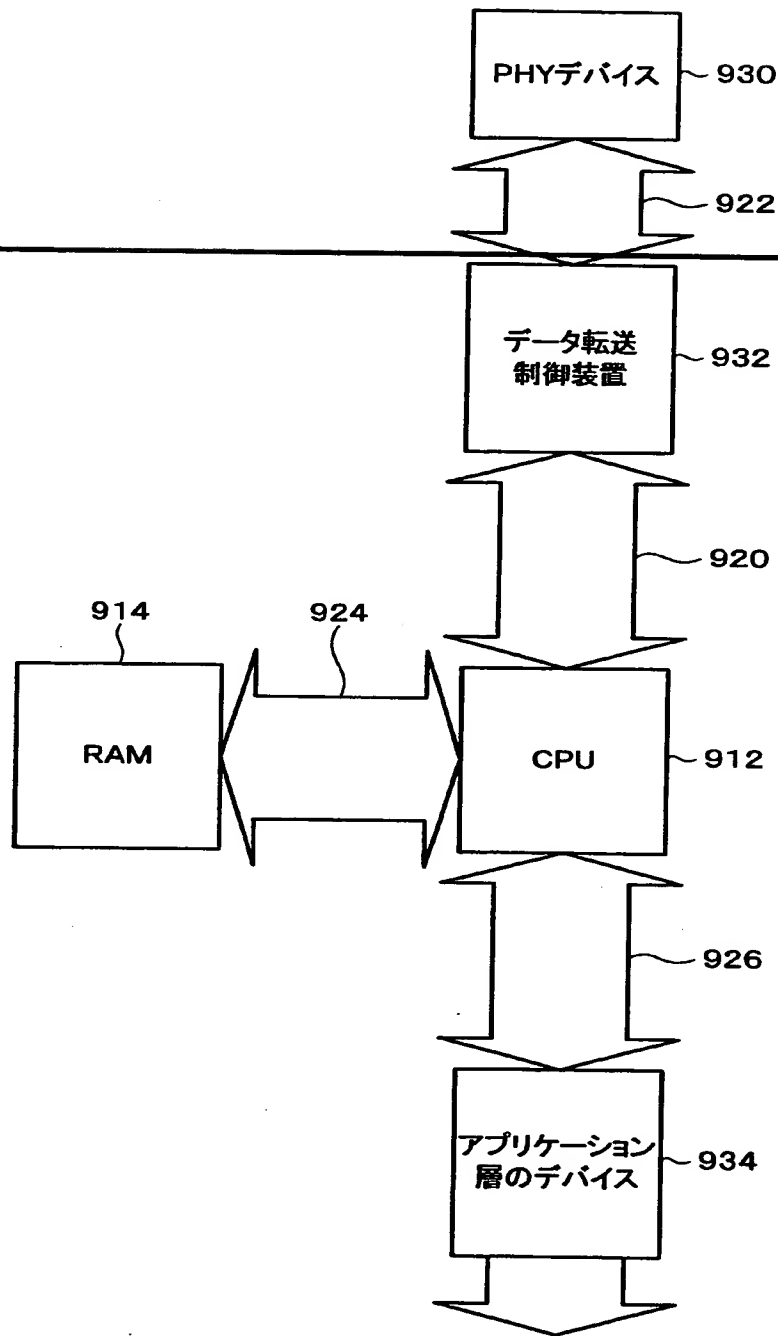
【図 5】



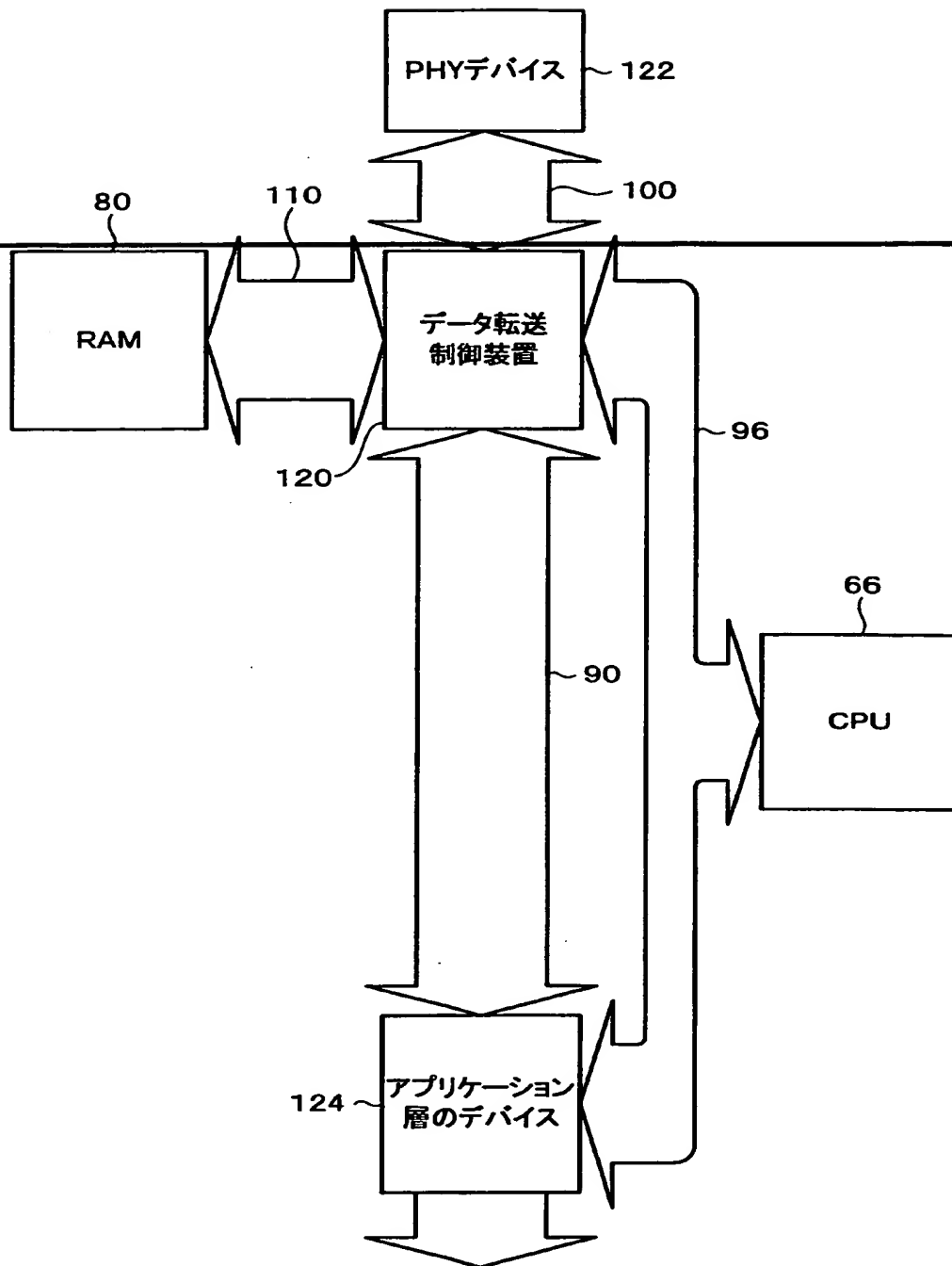
【図 6】



【図 7】

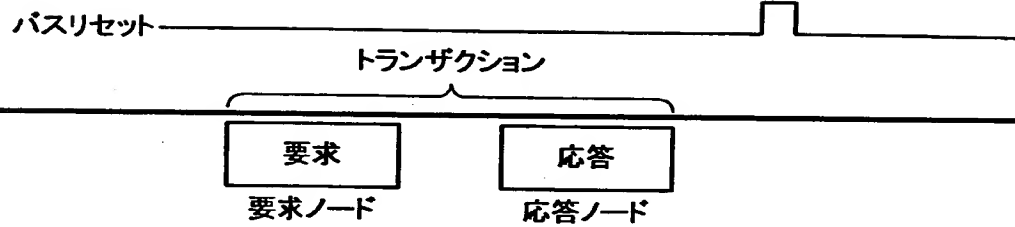


【図 8】

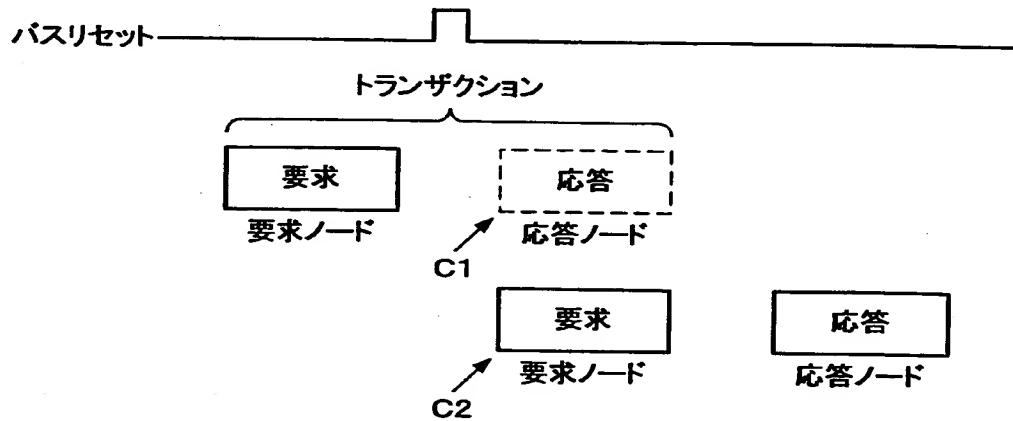


【図 9】

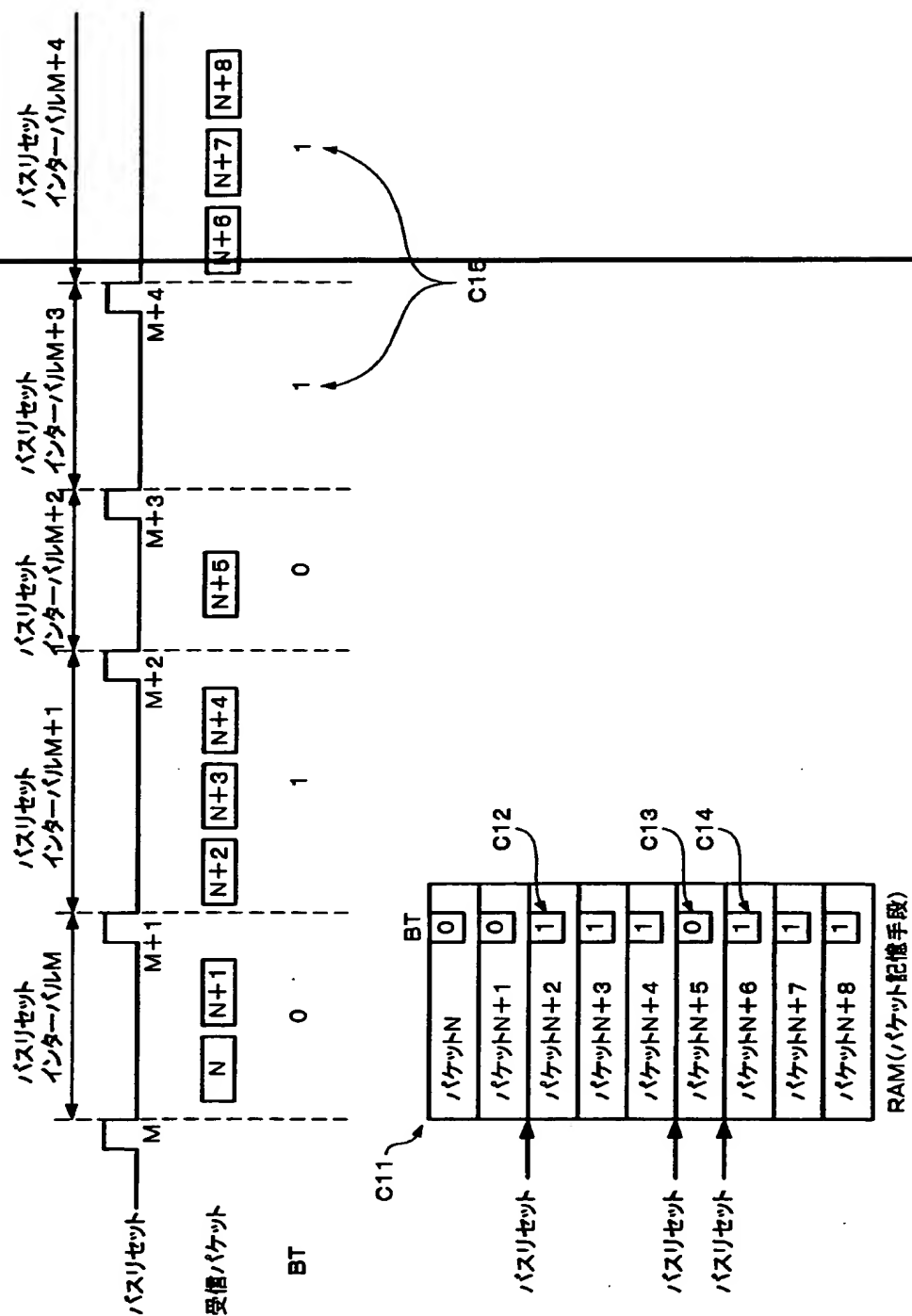
(A)



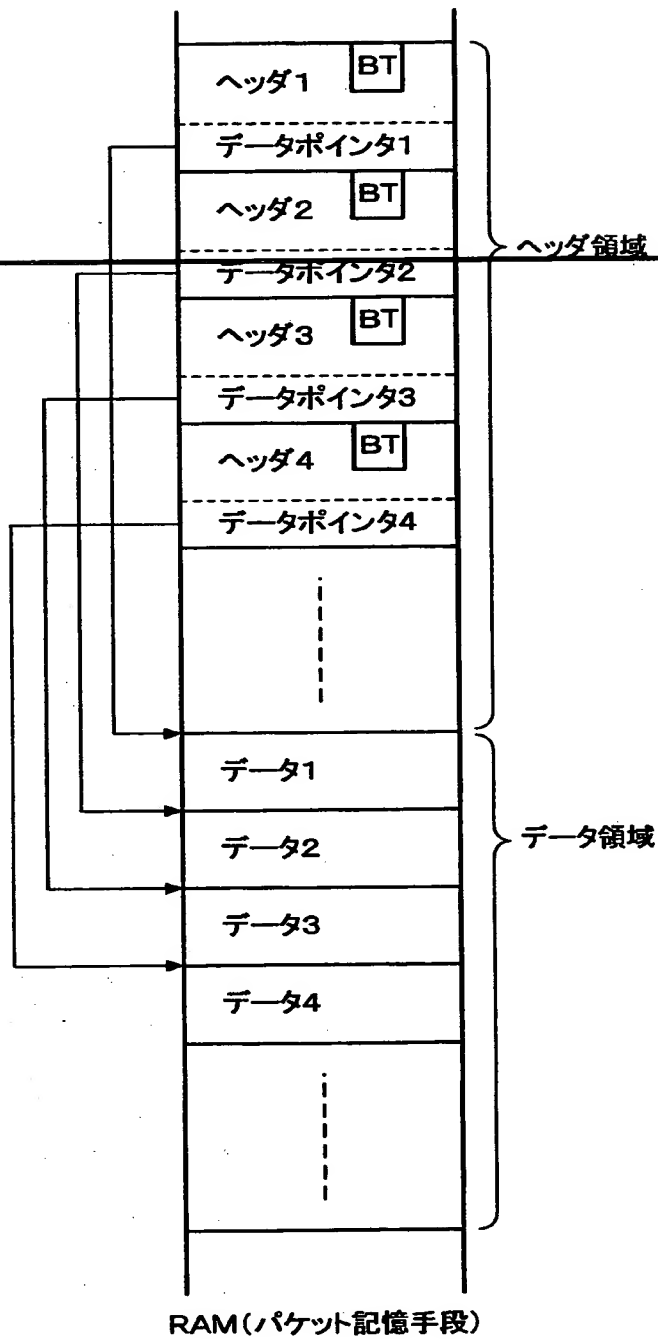
(B)



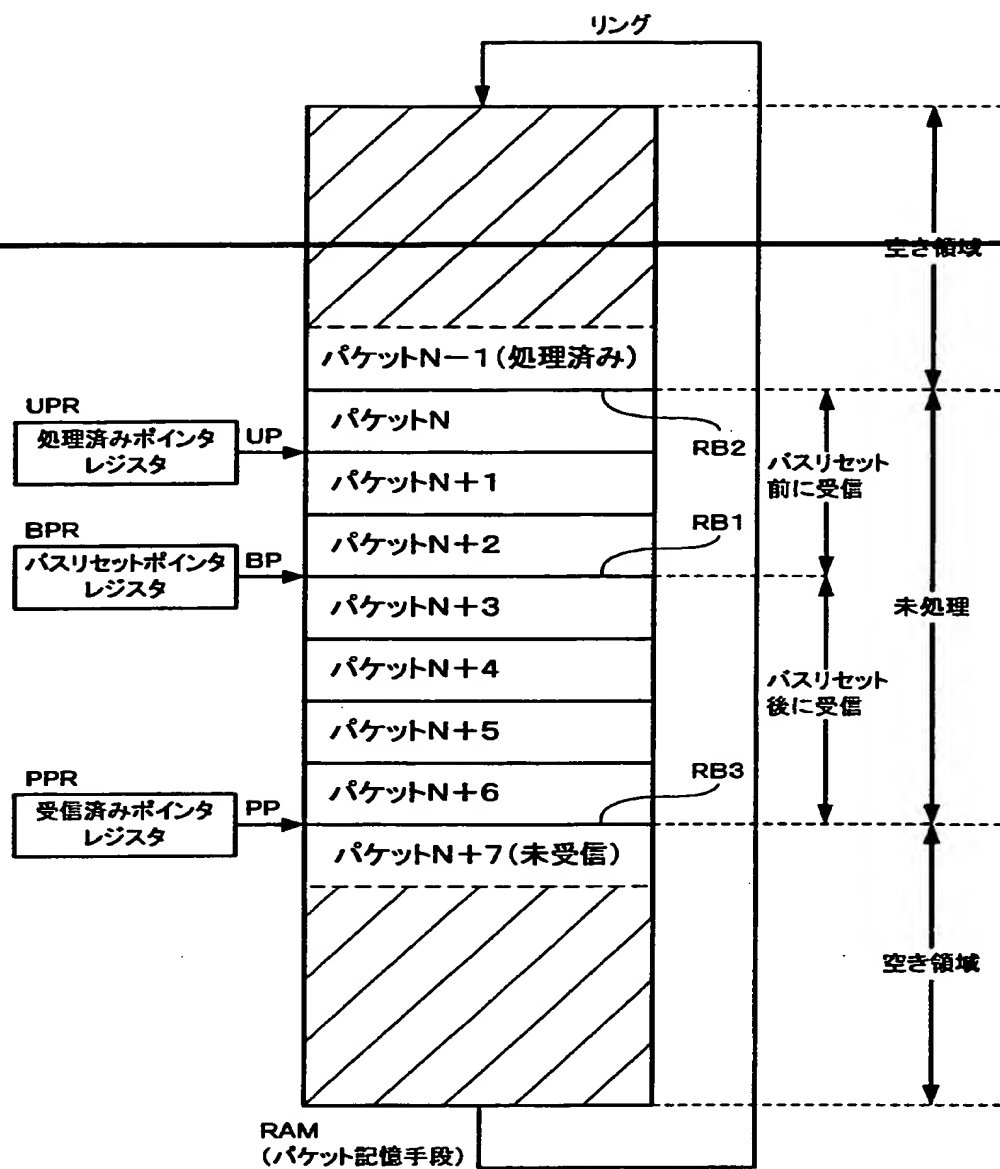
【図 10】



【図 11】

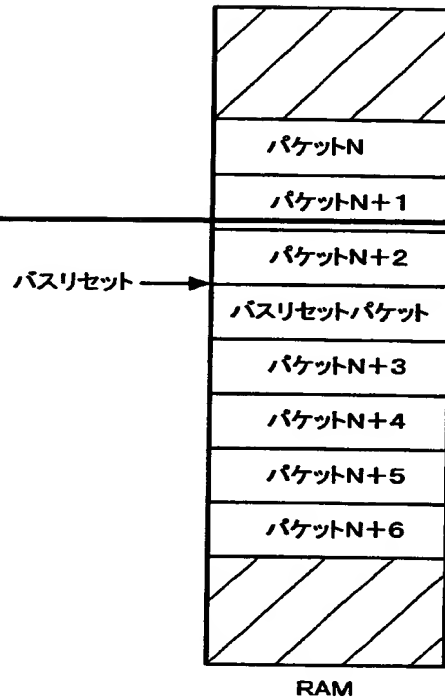


【図 12】

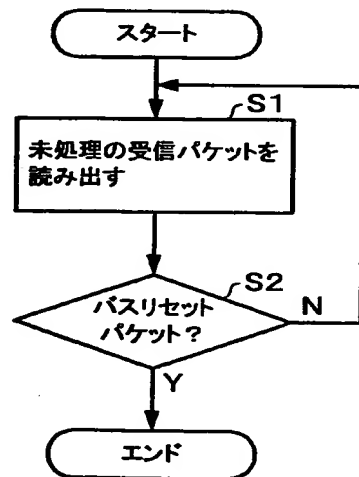


【図 13】

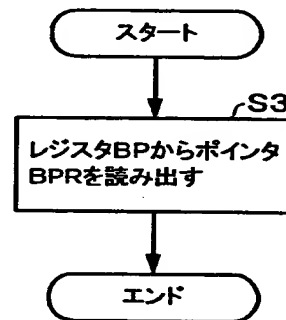
(A) 比較例



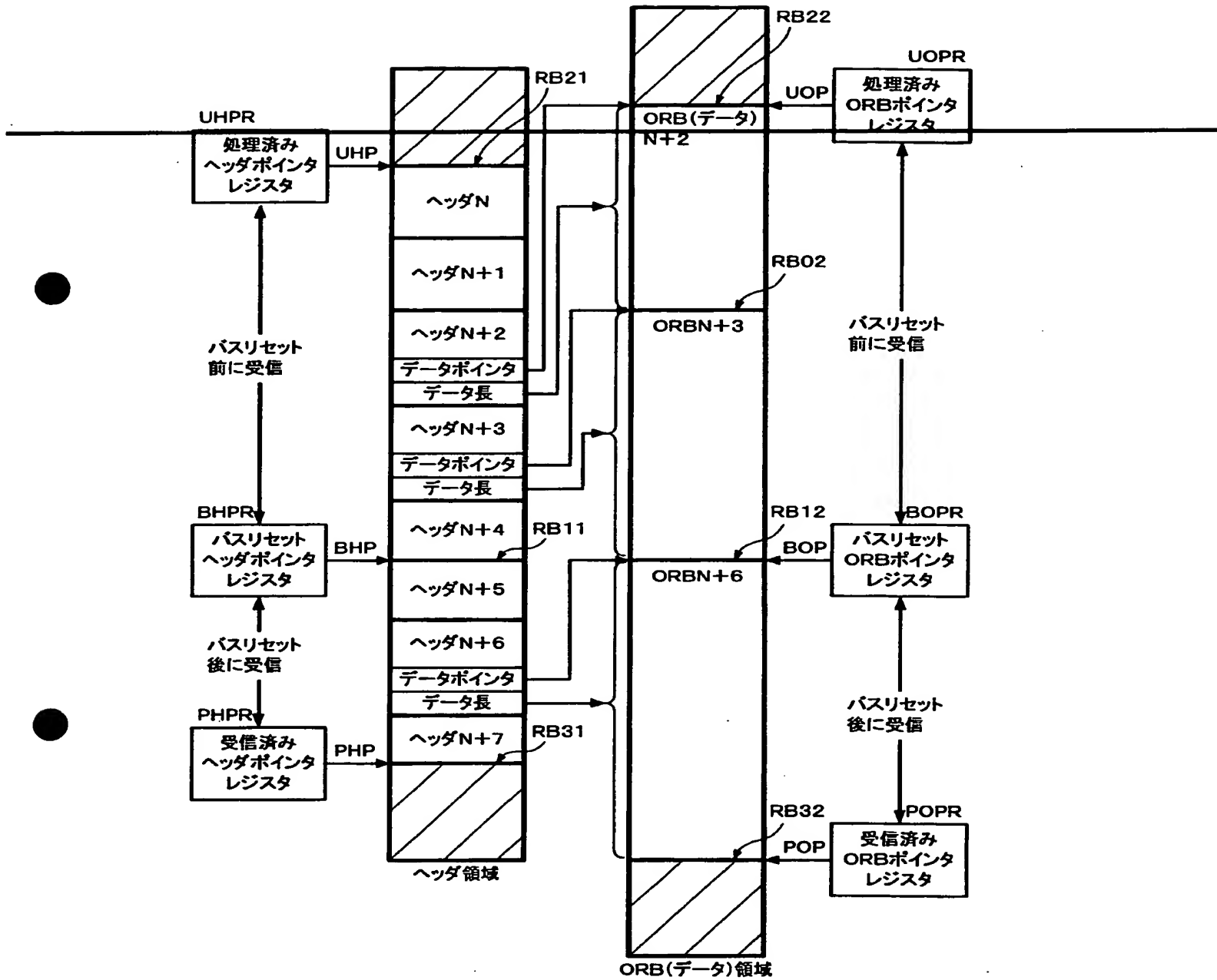
(B) バスリセットパケットを用いる手法



(C) バスリセットポイントを用いる手法

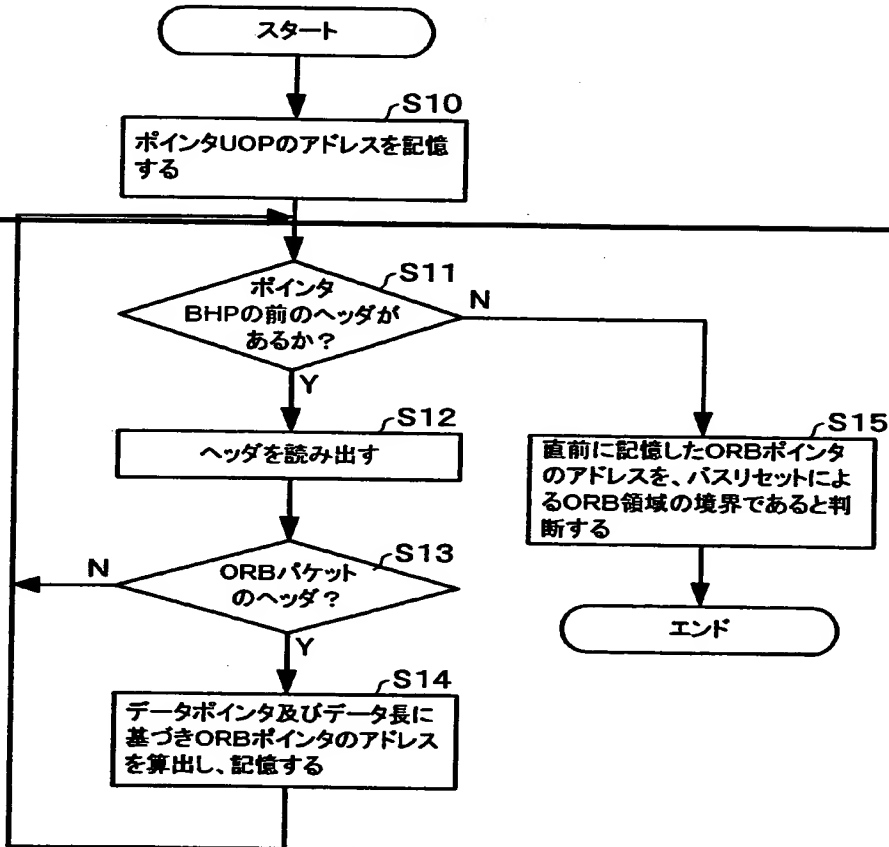


【図 14】

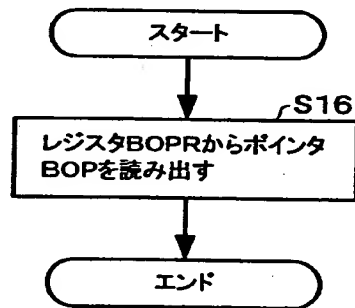


【図 15】

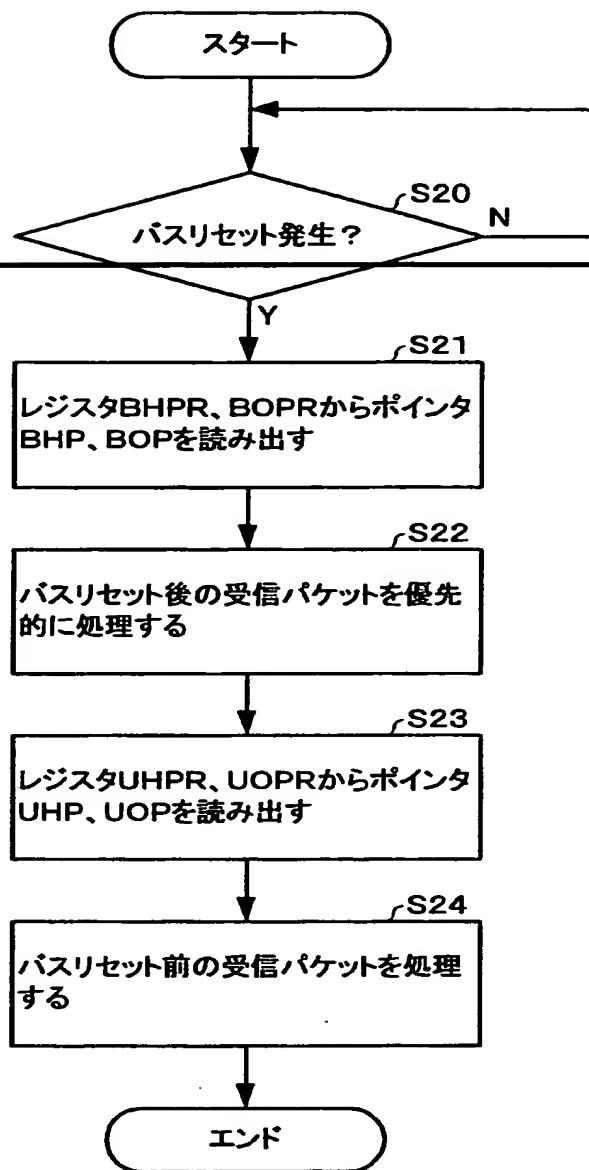
(A) バスリセットヘッダポイントのみを用いる手法



(B) バスリセットORBポイントを用いる手法

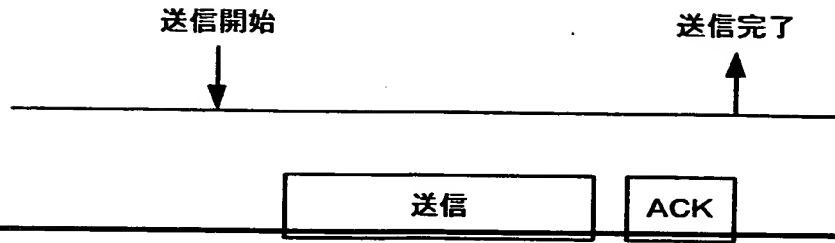


【図 16】

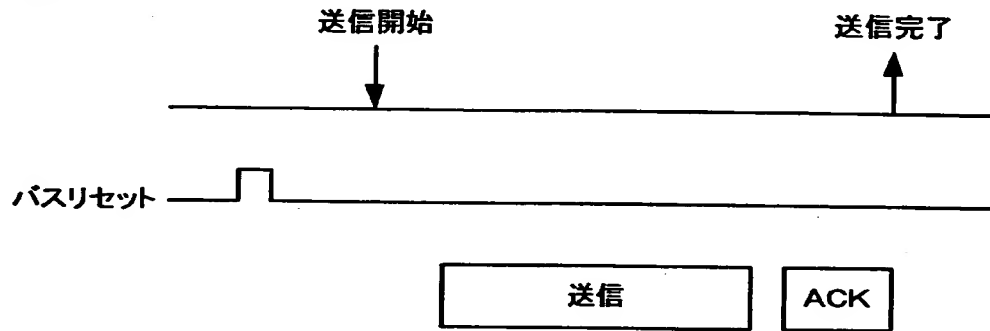


【図 17】

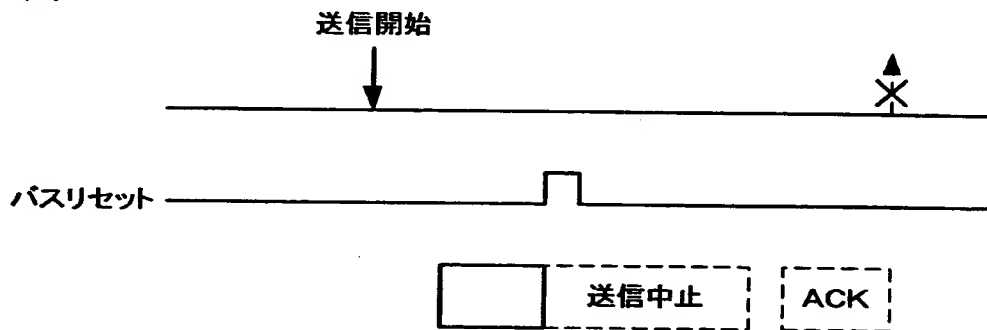
(A)



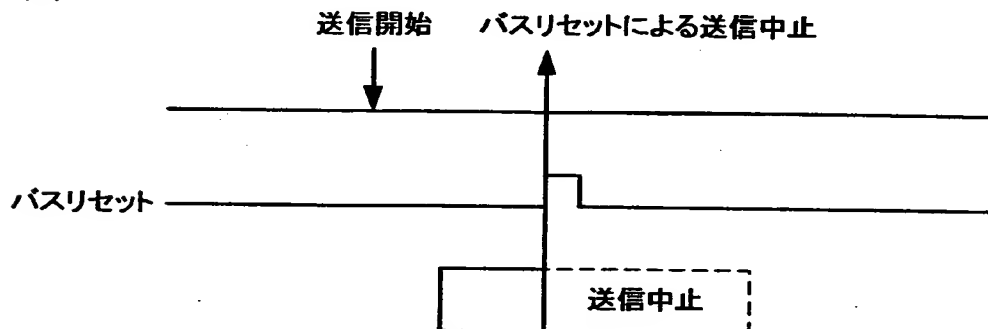
(B)



(C)

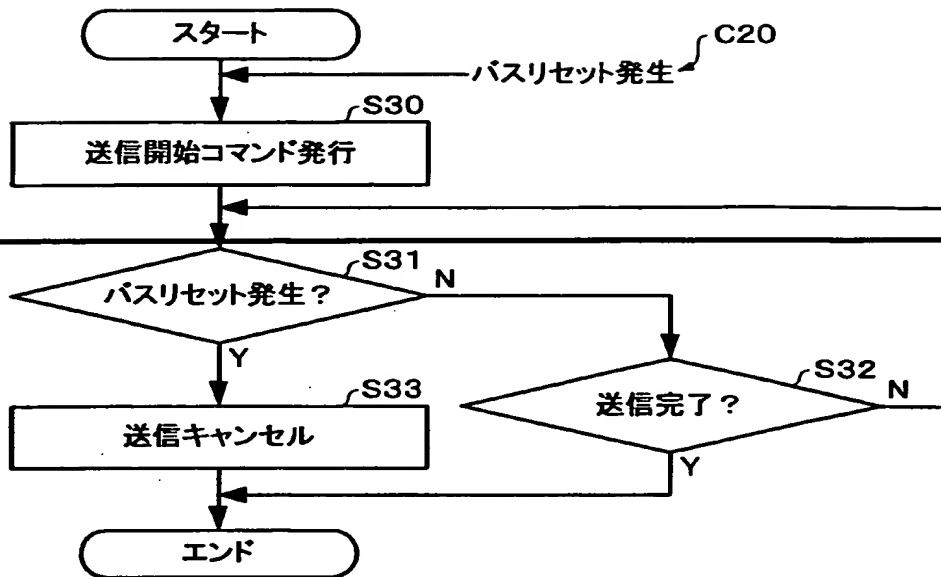


(D)

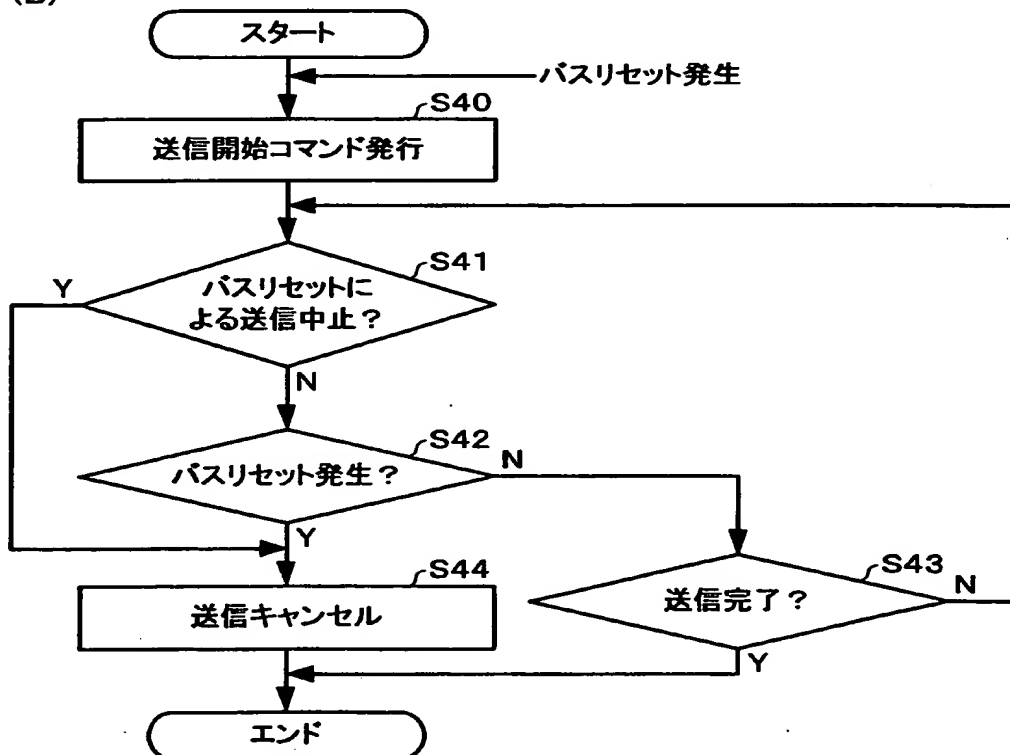


【図 18】

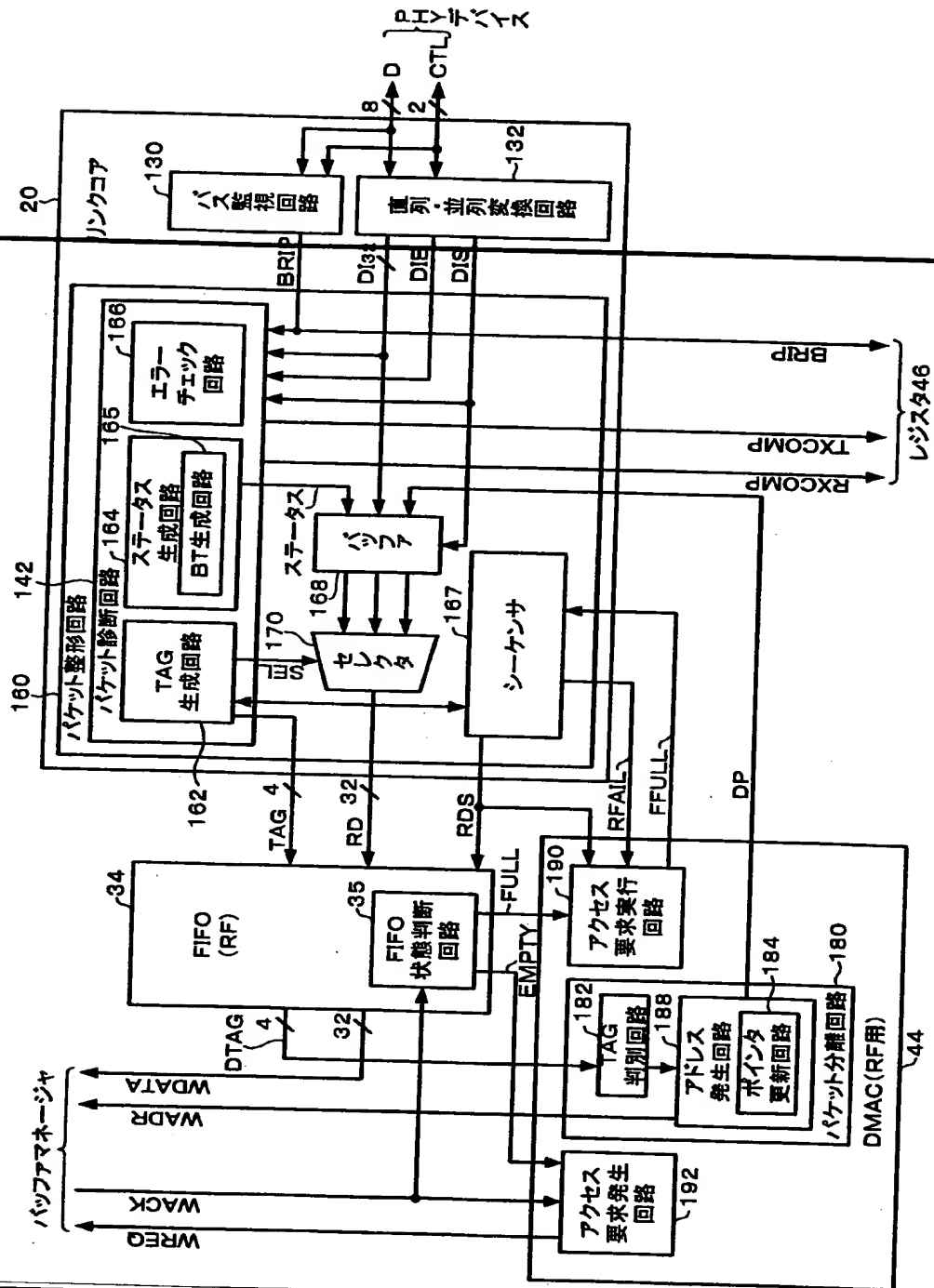
(A)



(B)



【图 19】



【図 2 0】

(A)

b.31							b.0
DestID	t1		rt	tcode	pri		(LSB)
SourceID	(MSB)						
PacketTypeSpecInfo							
DataLength	ExtendedTcode						
Header CRC							
Data							
Data CRC							

(B)

(B) RxAsynchronousPacket BlockWriteReq,BlockReadReq,LockReq,LockResp															
IEEE 1394															
b.31															
C31															
		speed		AS		BT		BG		O		ACK			
DestID				t1				rt		tcode				pri	
SourceID				(MSB)											
				PacketTypeSpecInfo								(LSB)			
DataLength				ExtendedTcode											
				DataPointer											
				reserved											
				C30								RAM			

【図 21】

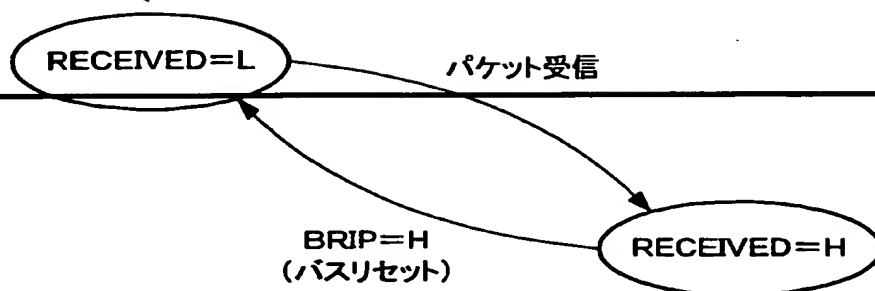
TAG(DTAG)	意味
0001	FW-SOP
0010	FW-HDR
0011	FW-FTR
0100	FW-ORB
0101	FW-STRM
1001	HW-SOP
1010	HW-HDR
1011	HW-FTR
1100	HW-ORB
1101	HW-STRM

FW ファームウェア
HW ハードウェア
SOP スタートオブパケット
 (受信パケットの最初の1クワドレット)
HDR SOP以外のヘッダ
FTR ACK送信
ORB ORB(データ)
STRM..... ストリーム(データ)

【図 22】

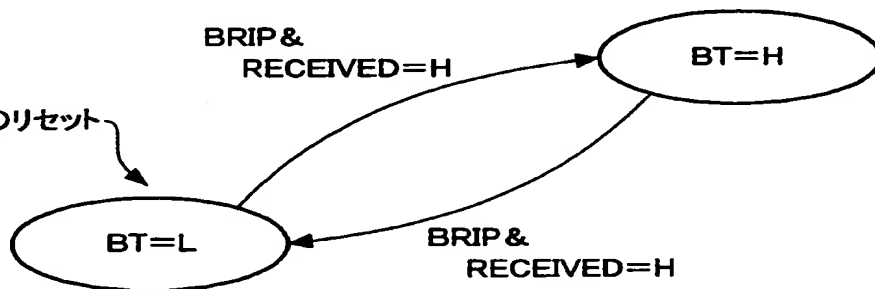
(A)

装置のリセット

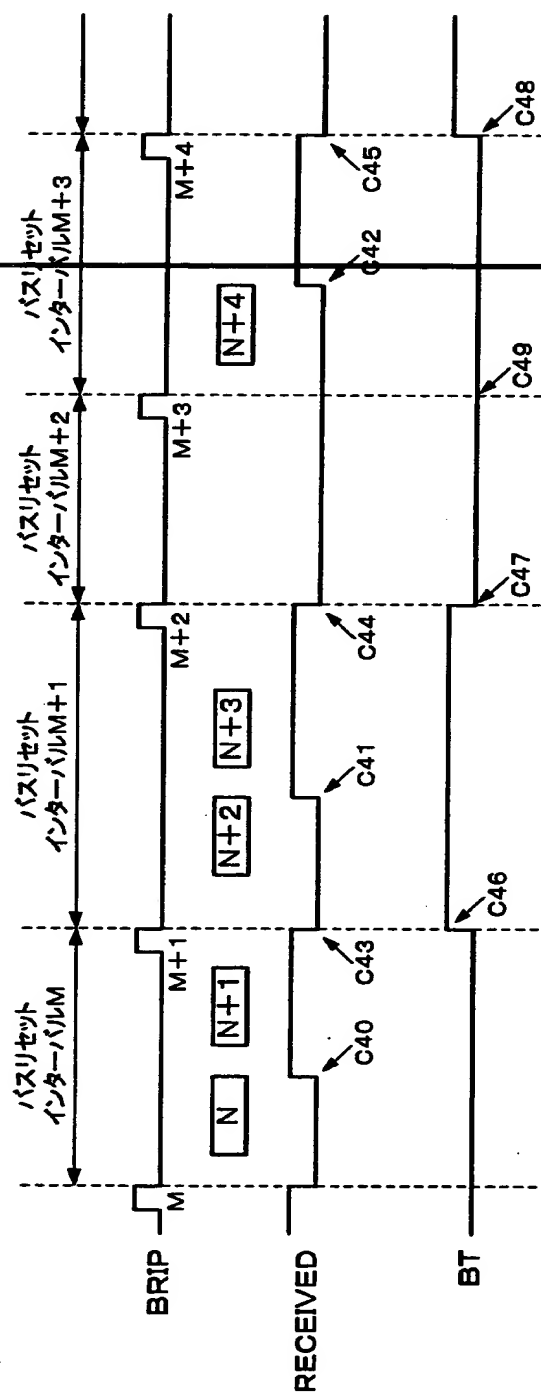


(B)

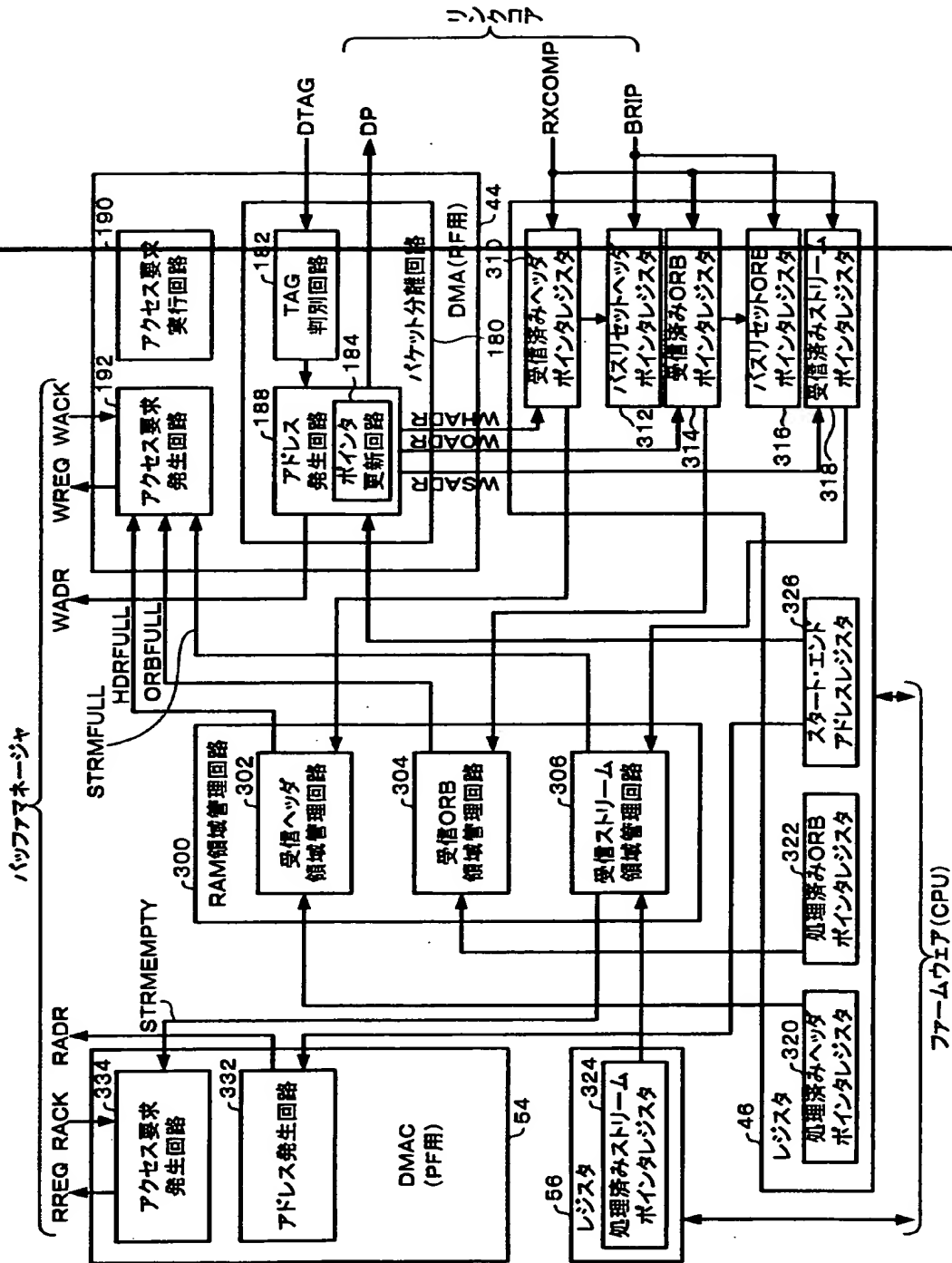
装置のリセット



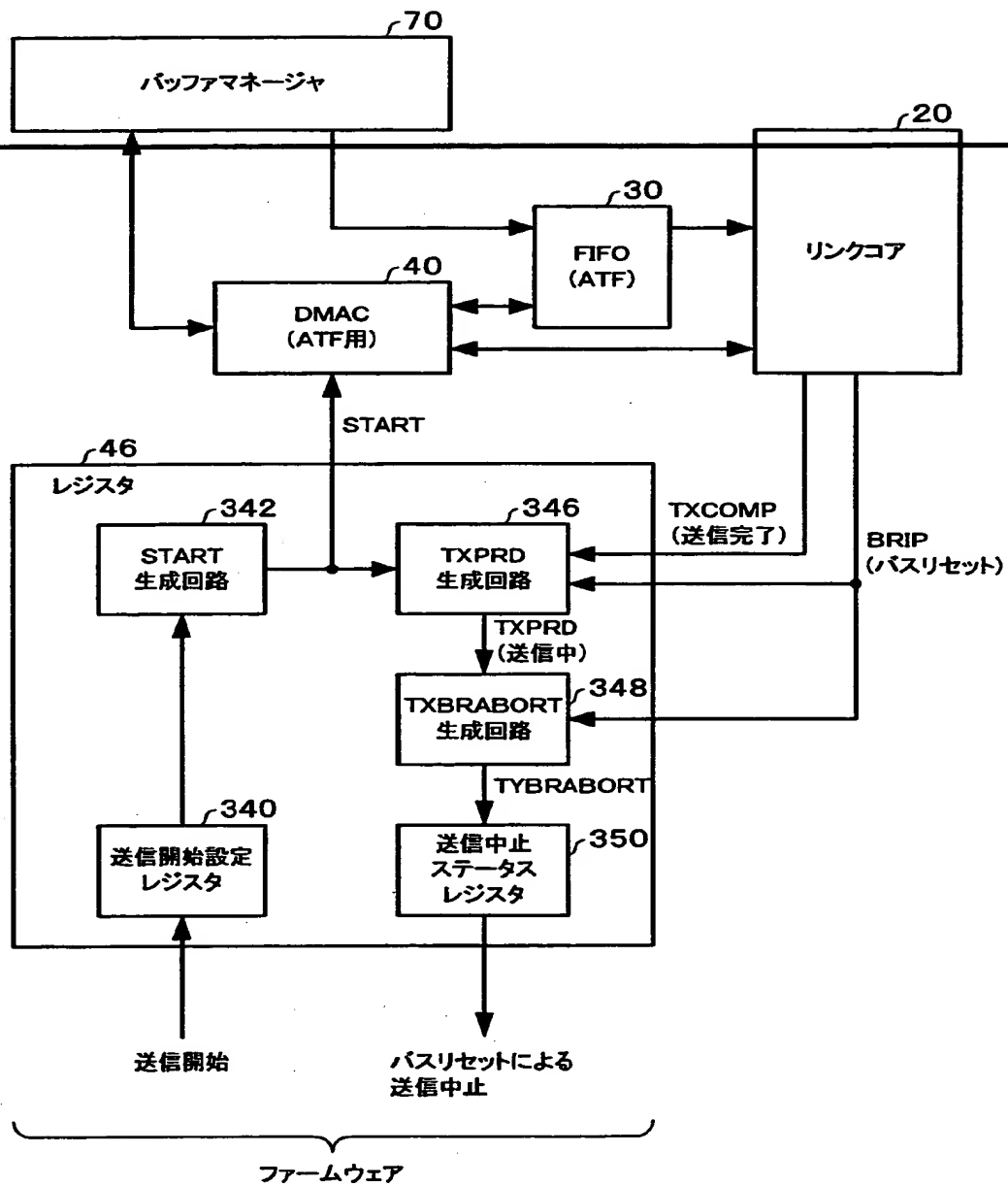
【図 23】



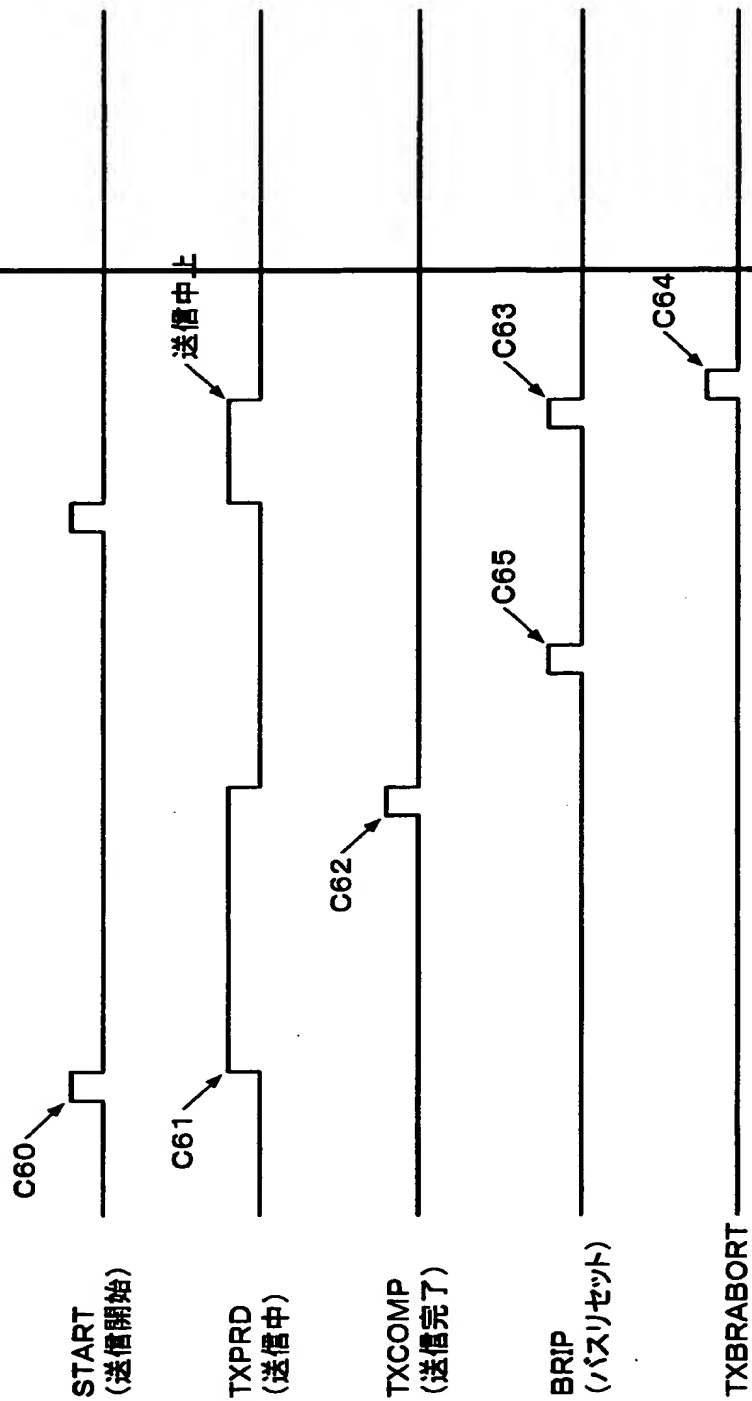
【図 2 4】



【図 25】

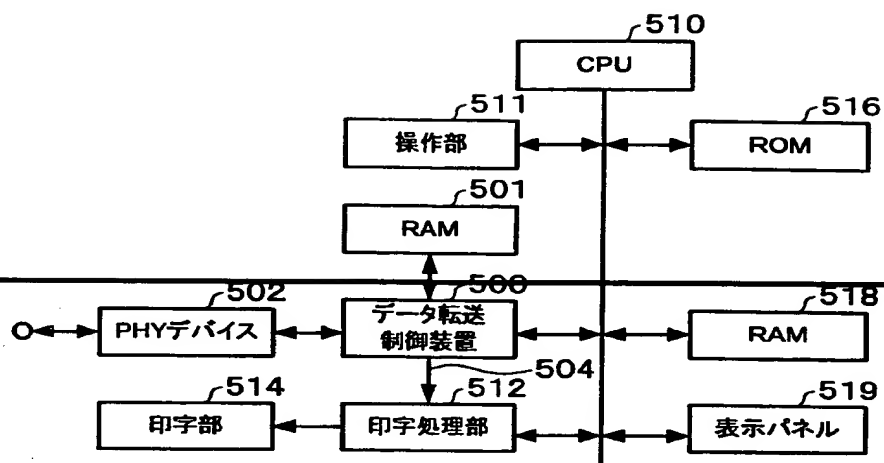


【図 2 6】

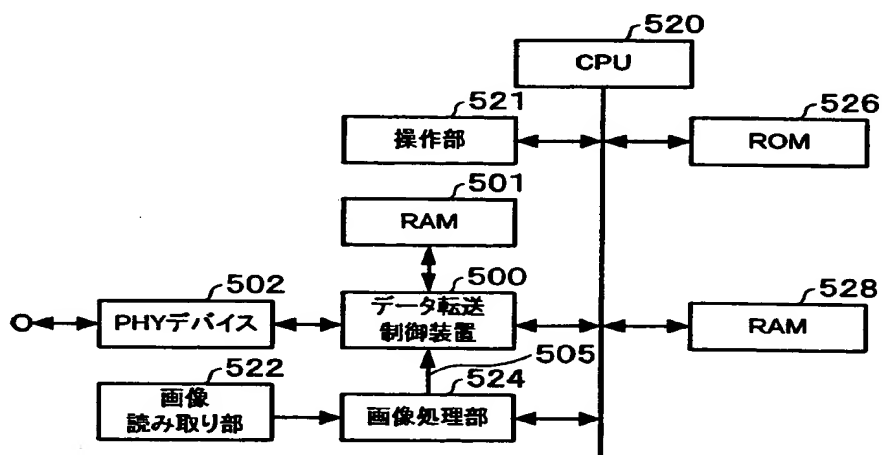


【図 27】

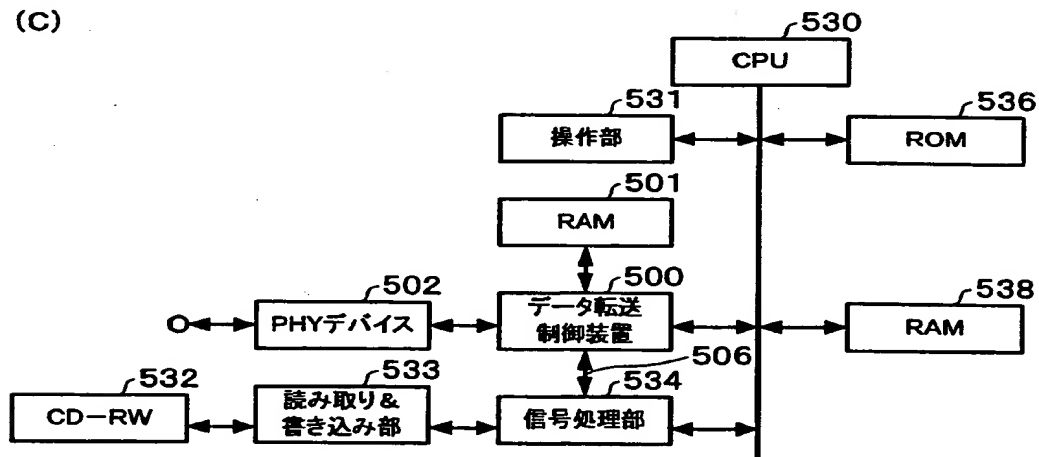
(A)



(B)

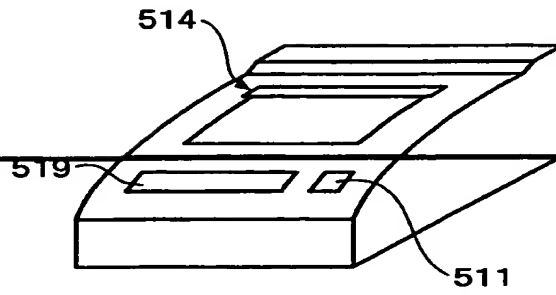


(C)

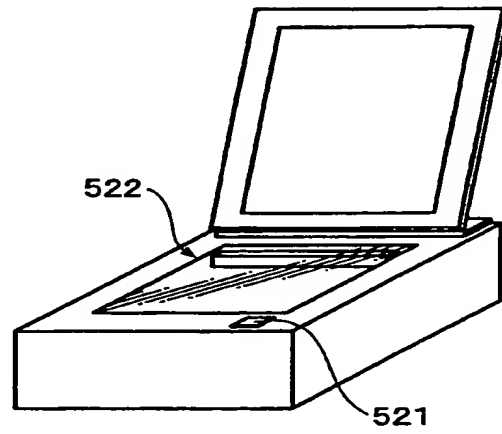


【図 28】

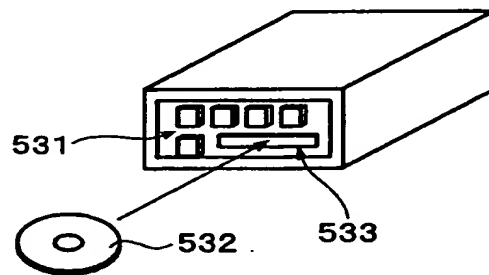
(A)



(B)



(C)



【書類名】 要約書

【要約】

【課題】 バスリセット発生時のファームウェアの処理負担を軽減できるデータ転送制御装置及びこれが用いられる電子機器を提供することにある。

【解決手段】 IEEE 1394 規格のデータ転送制御装置において、受信パケットと次の受信パケットとが異なるバスリセットインターバルに受信したパケットである場合にトグルするビット BT を生成し、この BT を、RAM に記憶される各パケットの各ヘッダに含ませる。RAM 上でのバスリセットの境界を指すバスリセットポインタ（バスリセットヘッダポインタ、バスリセット ORB ポインタ）を用意し、バスリセットの発生前の受信パケットと発生後の受信パケットとを容易に区別できるようにする。バスリセットの発生により送信が中止された場合に、バスリセット送信中止ステータスをレジスタを介してファームウェアに伝える。

【選択図】 図 10

出 願 人 履 歴 情 報

識別番号 [000002369]

1. 変更年月日 1990年 8月20日

[変更理由] 新規登録

住 所 東京都新宿区西新宿2丁目4番1号

氏 名 セイコーエプソン株式会社

THIS PAGE BLANK (USPTO)